The Changing Face of ETL

Event-Driven Architectures for Data Engineers

@rmoff

Photo by Samuel Sianipar on Unsplash

Photo by Khai Sze Ong on Unsplash

Photo by Ramzer Ridzo on Unsplash

Photo by Rohit Tandon on Unsplash

Photo by Theodore Moore on Unsplash

Photo by Cristian Grecu on Unsplash

It used to be so simple

Photo by Patrick Fore on Unsplash

More Sources

Photo by Eugenio Mazzone on Unsplash

Photo by Tom Barrett on Unsplash

Photo by Kirill on Unsplash

Batches and Buckets

Applications

Respond

→ an order was placed!

Analytics

Tell Us What Happened

→ how many orders were placed

Photo by Deva Darshan from Pexels

Photo by NASA on Unsplash

$ whoami

- Robin Moffatt (@rmoff)
- Senior Developer Advocate at Confluent

  (Apache Kafka, not Wikis 😉)

- Working in data & analytics since 2001
- Oracle ACE Director (Alumnus)

http://rmoff.dev/**talks** • http://rmoff.dev/**blog** • http://rmoff.dev/**youtube**

Events

Photo by Mark Kamalov on Unsplash

An event is both:

- Notification
- State transfer

A Customer Experience

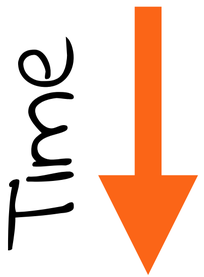
A Sensor Reading

Databases

The Stream/Table Duality

Stream

Account ID	Amount
12345	+ €50
12345	+ €25
12345	-€60



Table

Account ID	Balance
12345	€50

Account ID	Balance
12345	€75

Account ID	Balance
12345	€15

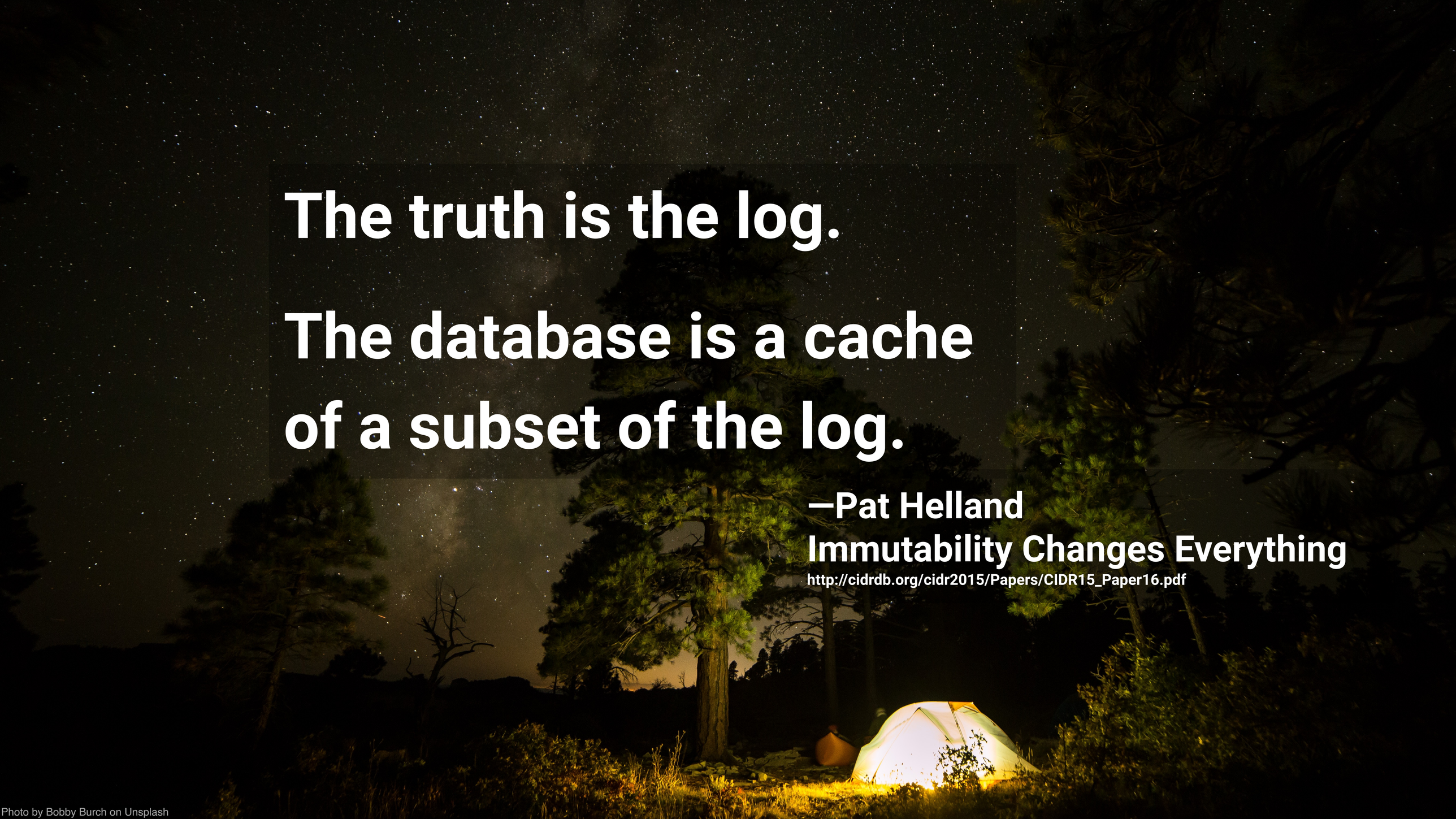
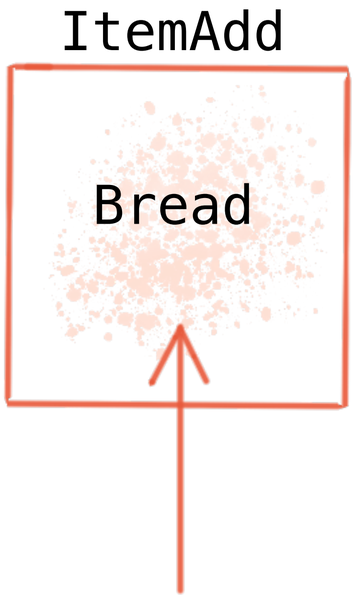
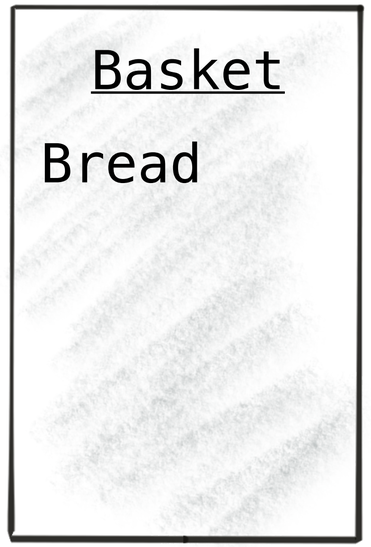


Photo by Bobby Burch on Unsplash

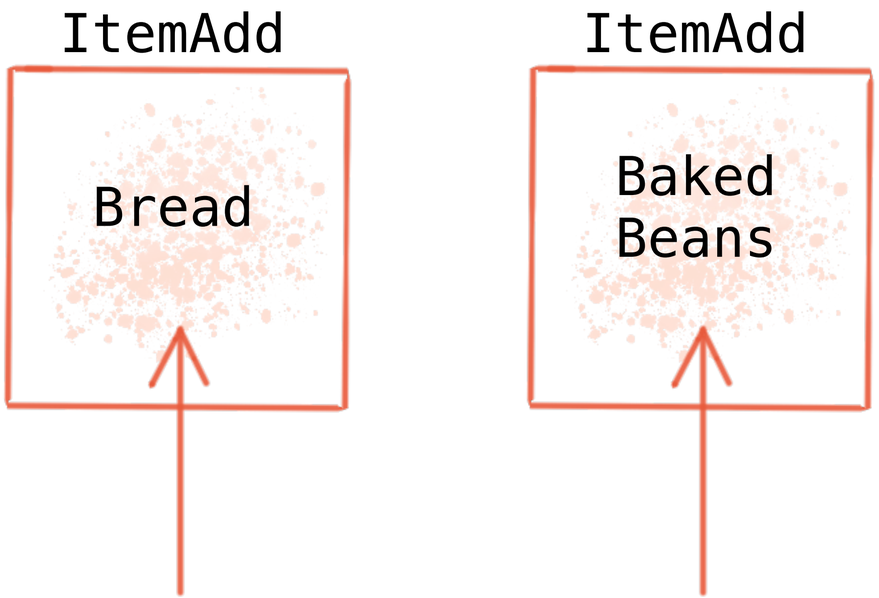
Events

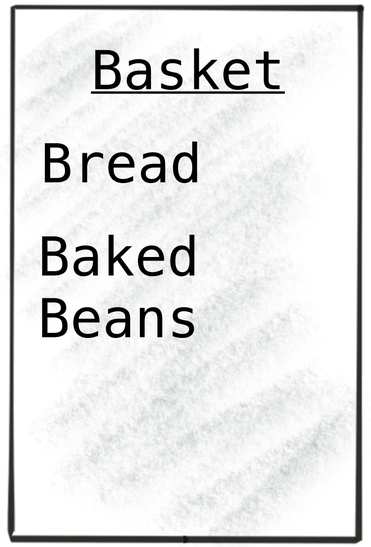
Events



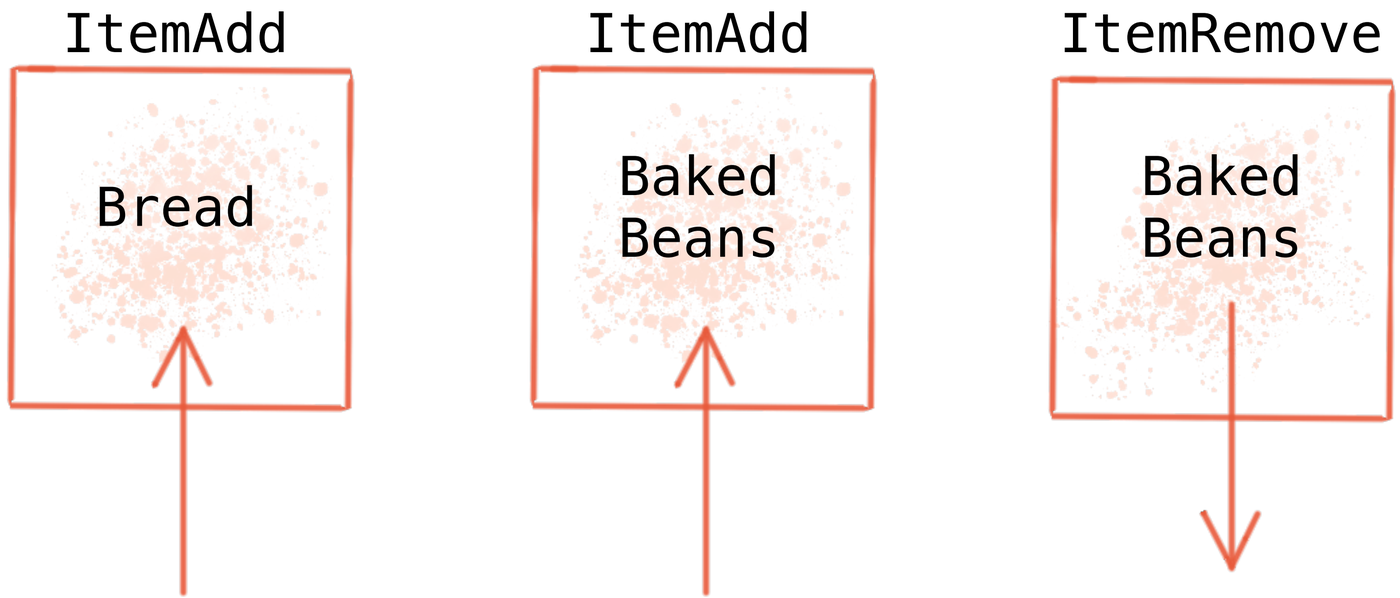


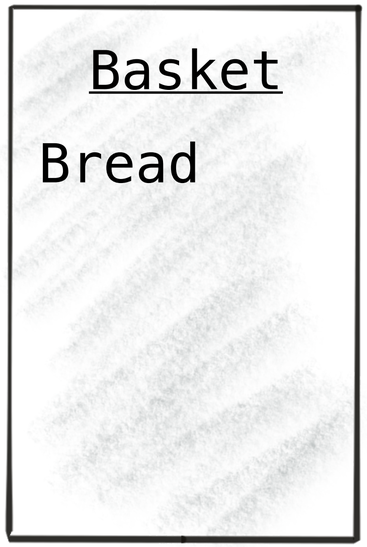
Events


ItemAdd
Bread
ItemAdd
Baked
Beans

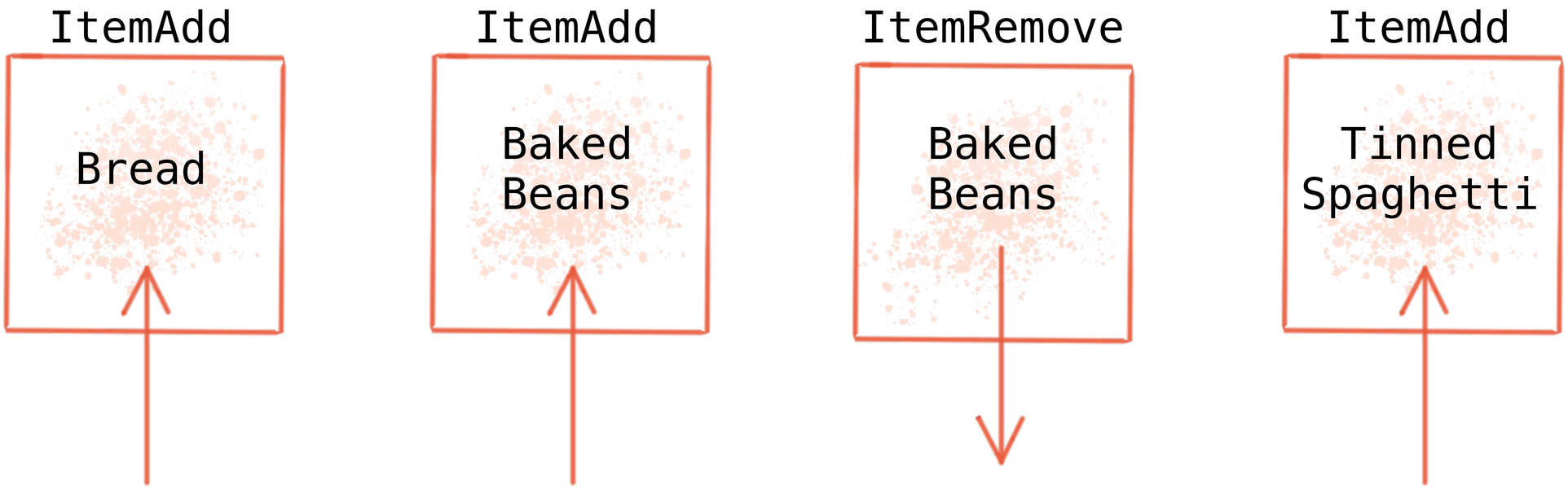

Basket
Bread
Baked
Beans

Events


ItemAdd
Bread
ItemAdd
Baked
Beans
ItemRemove
Baked
Beans


Basket
Bread

Events

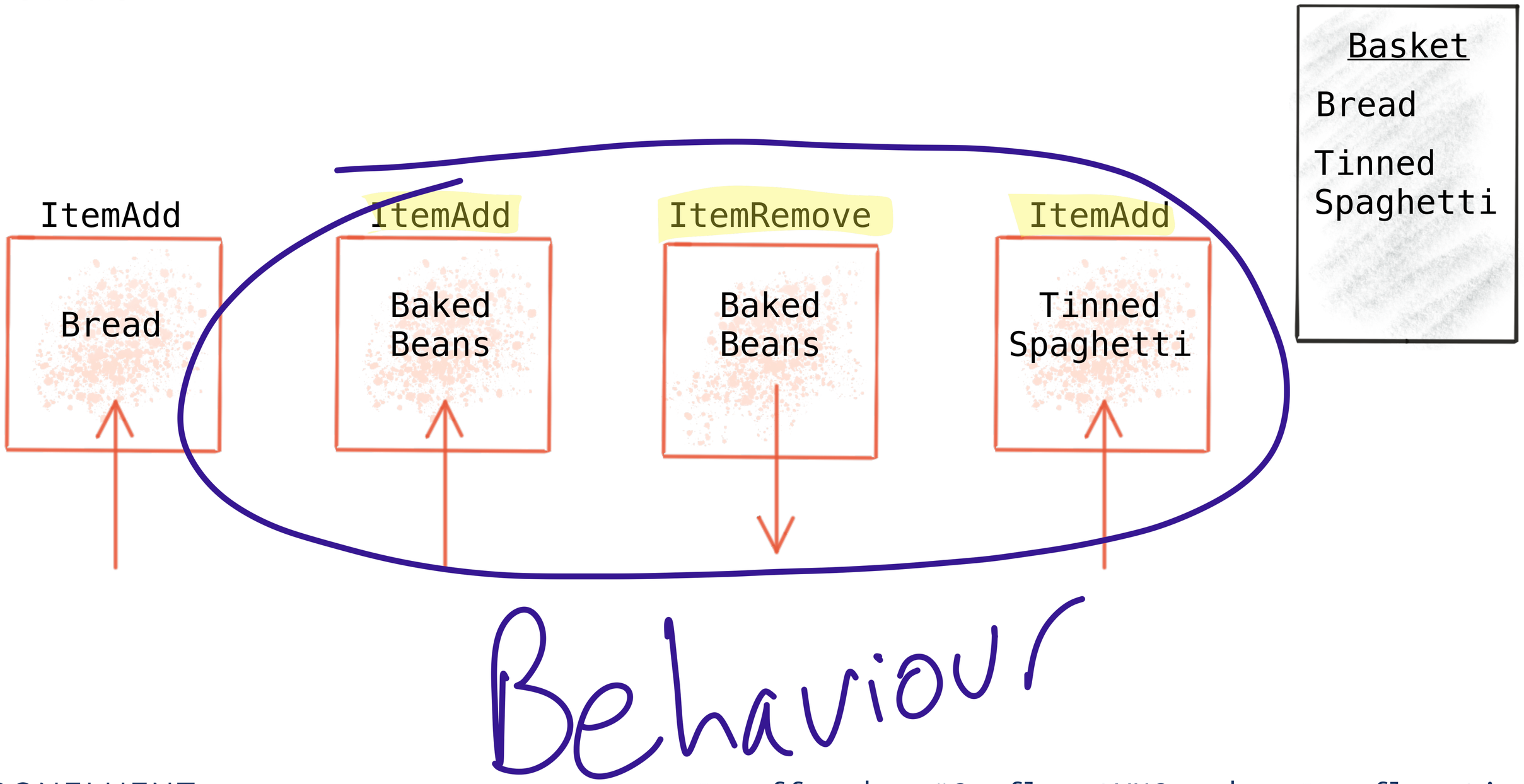


Basket

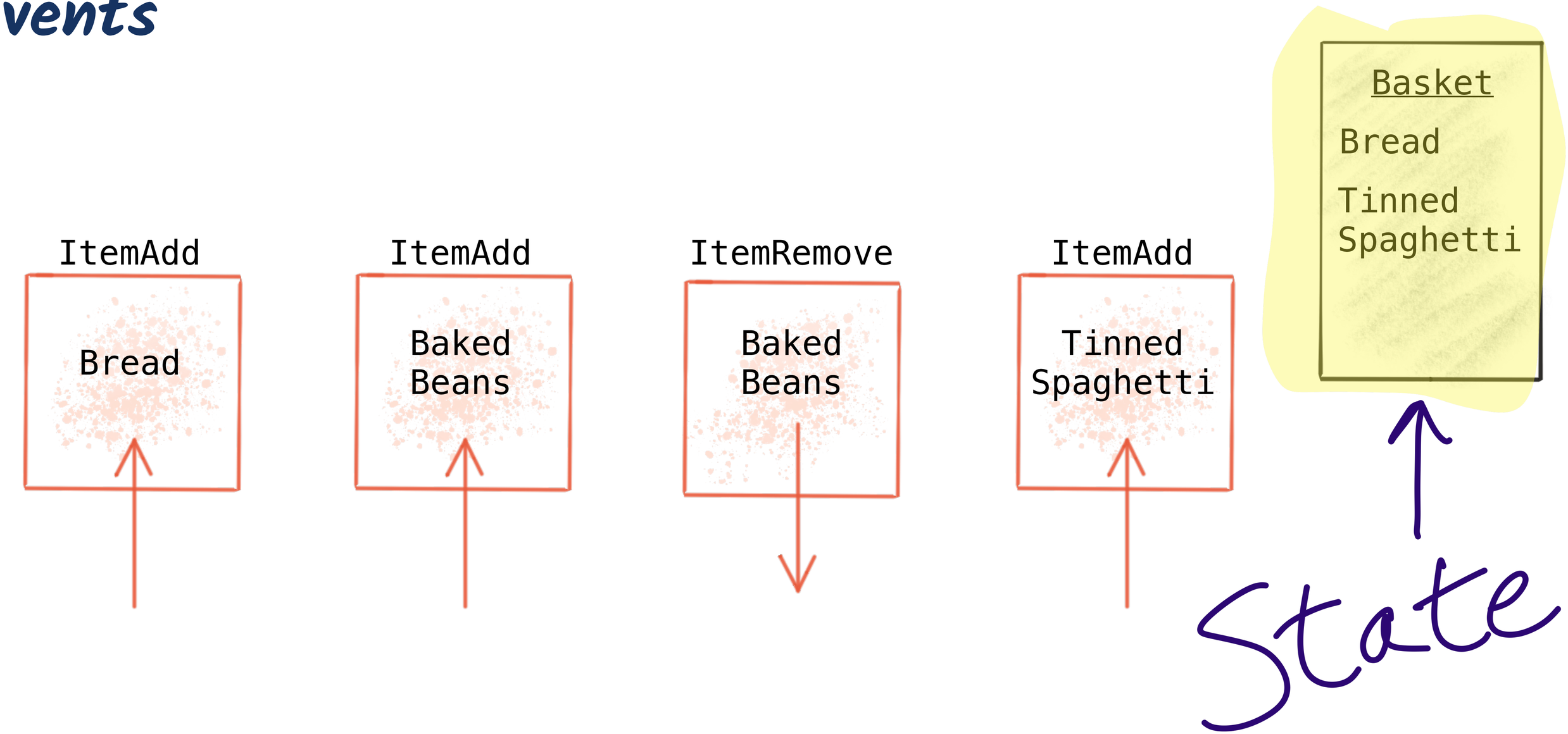
Bread

Tinned
Spaghetti

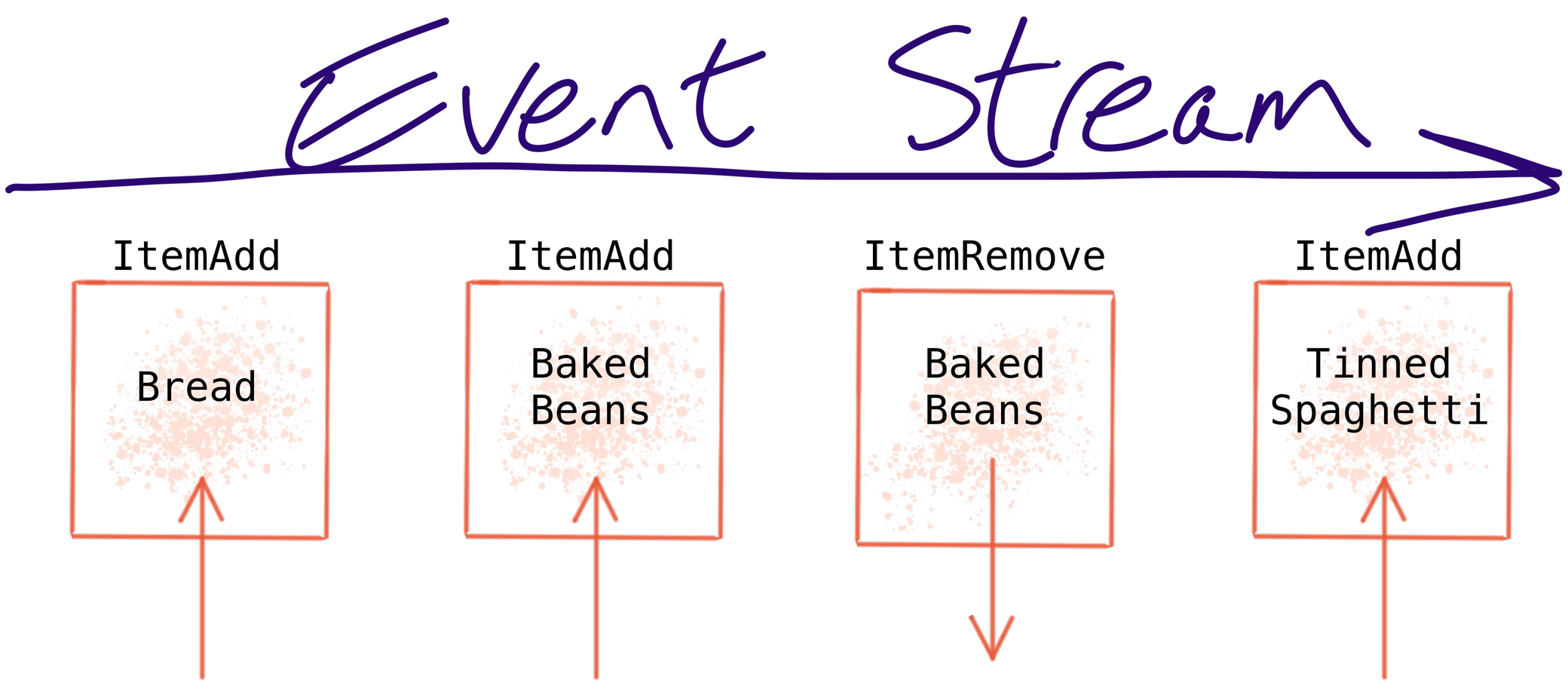
Events


Basket
Bread
Tinned Spaghetti
ItemAdd
Bread
ItemAdd
Baked Beans
ItemRemove
Baked Beans
ItemAdd
Tinned Spaghetti
Behaviour

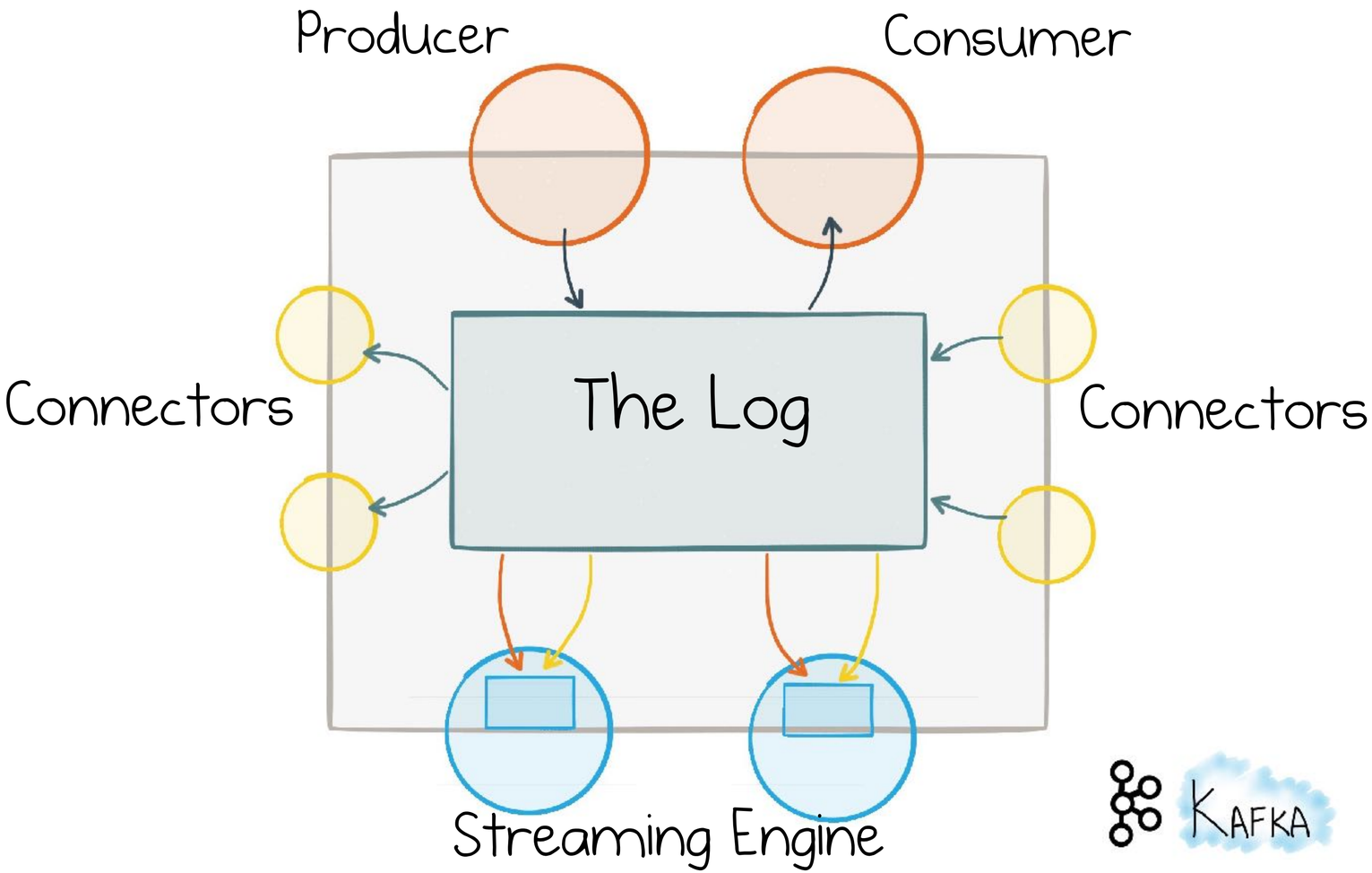
Events


ItemAdd
Bread
ItemAdd
Baked
Beans
ItemRemove
Baked
Beans
ItemAdd
Tinned
Spaghetti
Basket
Bread
Tinned
Spaghetti
State

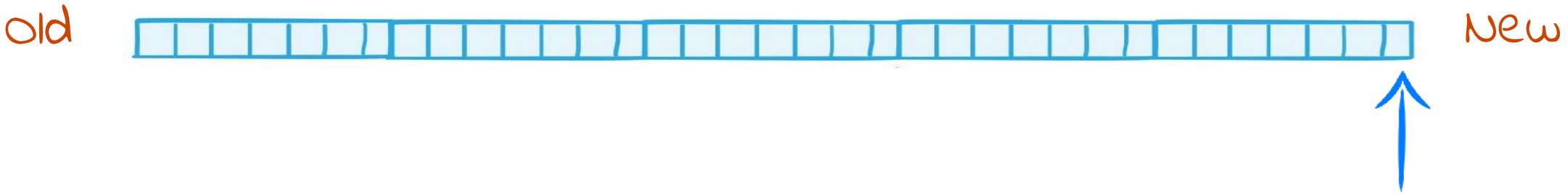
Events


Event Stream
ItemAdd
Bread
ItemAdd
Baked Beans
ItemRemove
Baked Beans
ItemAdd
Tinned Spaghetti
Basket
Bread
Tinned Spaghetti

What is an Event Streaming Platform?

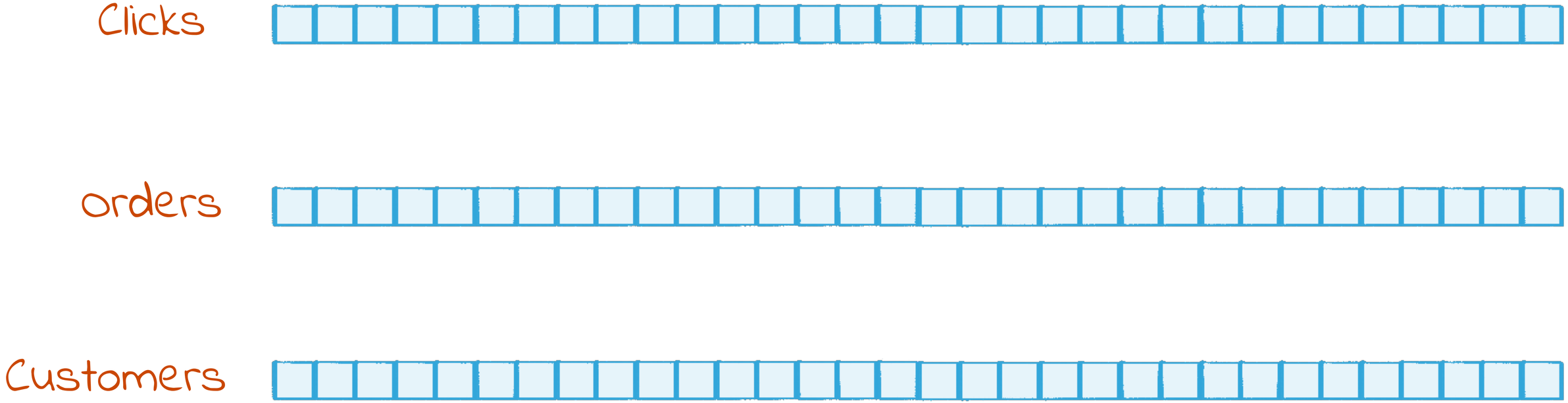


Immutable Event Log



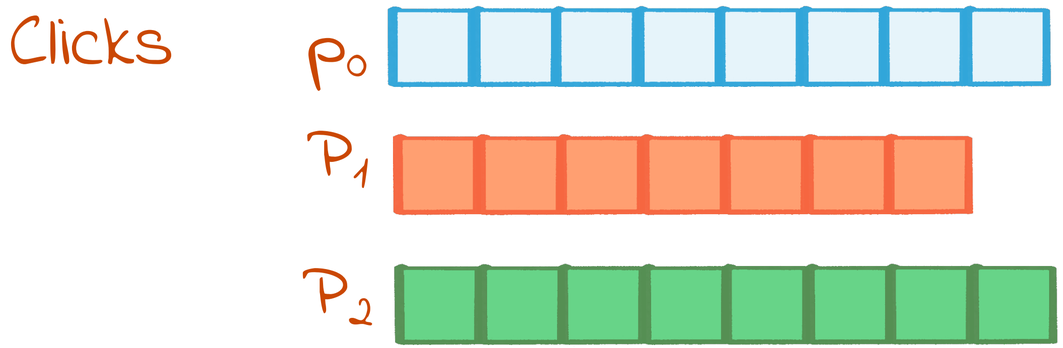
Messages are added at the end of the log

Topics



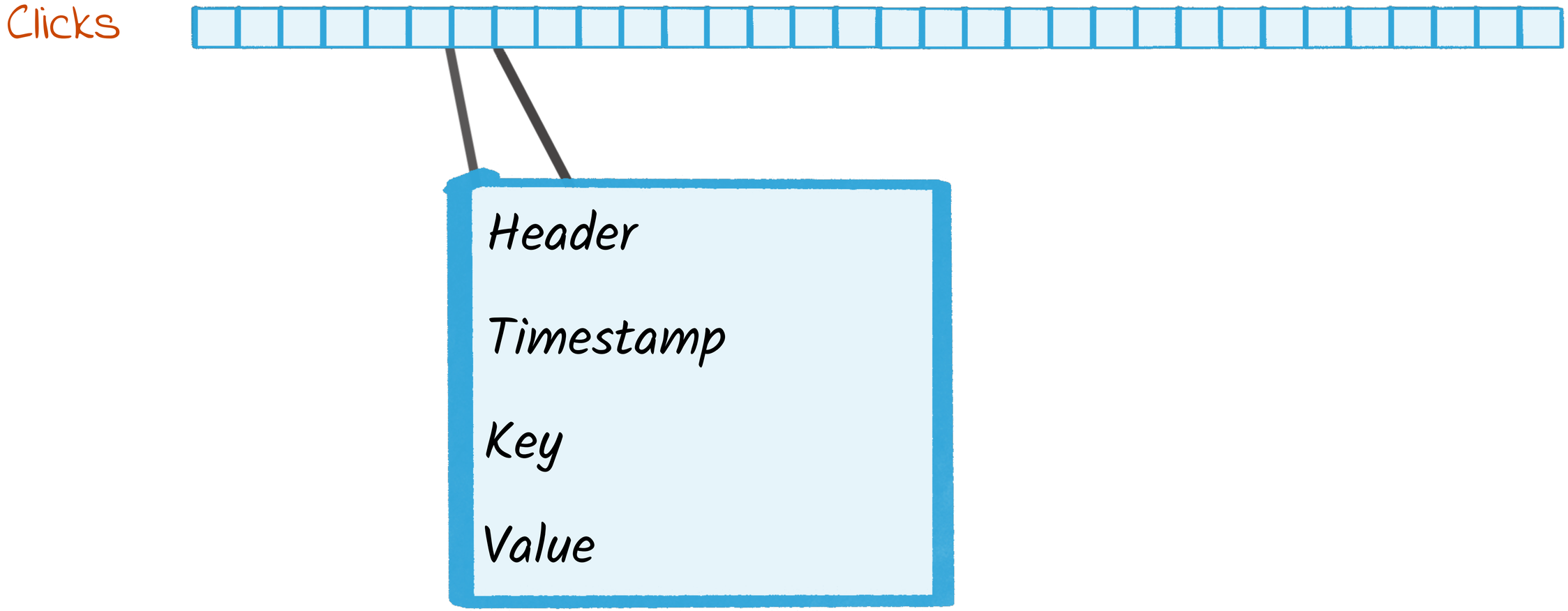
Topics are similar in concept to tables in a database

Partitions



Messages are guaranteed to be strictly ordered within a partition

Messages are just KN bytes *plus headers + timestamp*



Serialisation & Schemas

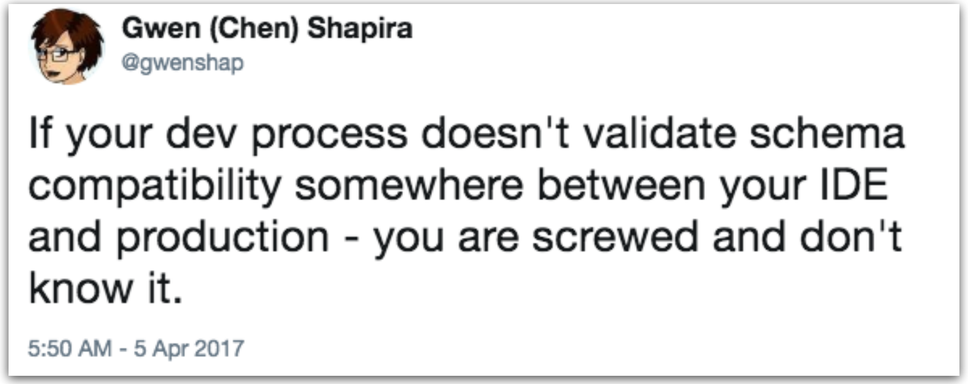
Avro 👍

Protobuf 👍

JSON Schema 👍

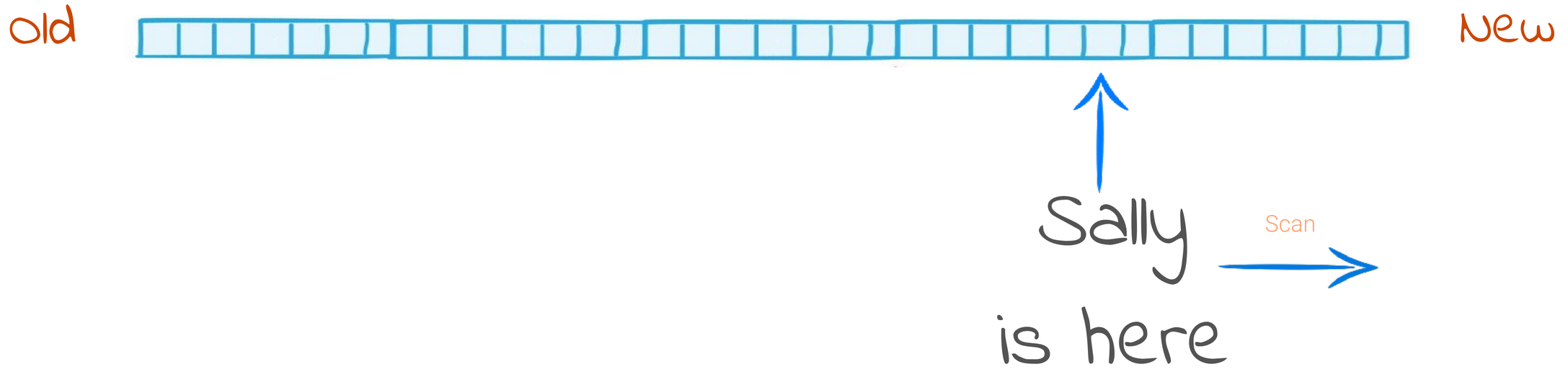
JSON 😬

CSV

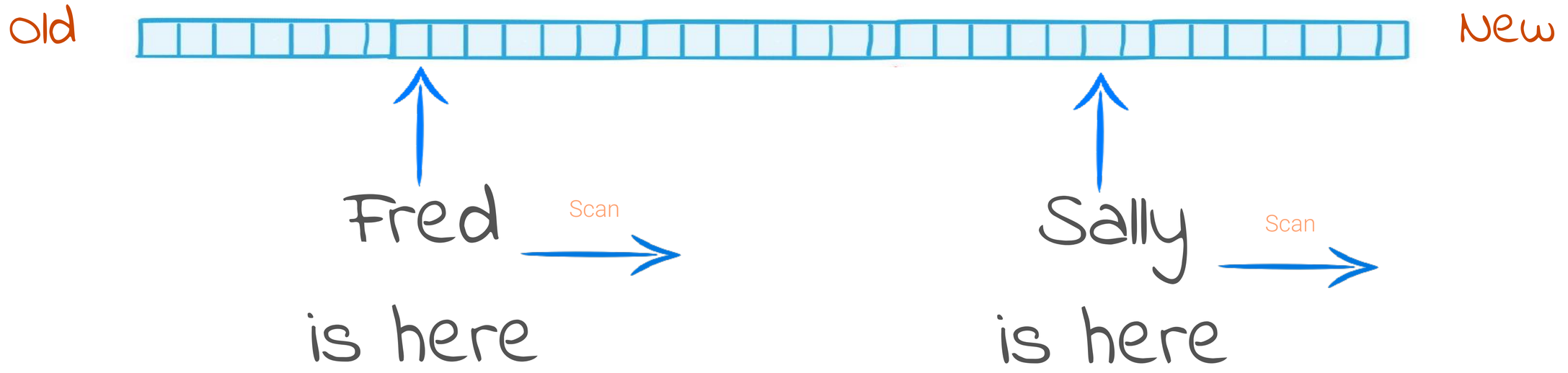


https://rmoff.dev/qcon-schemas

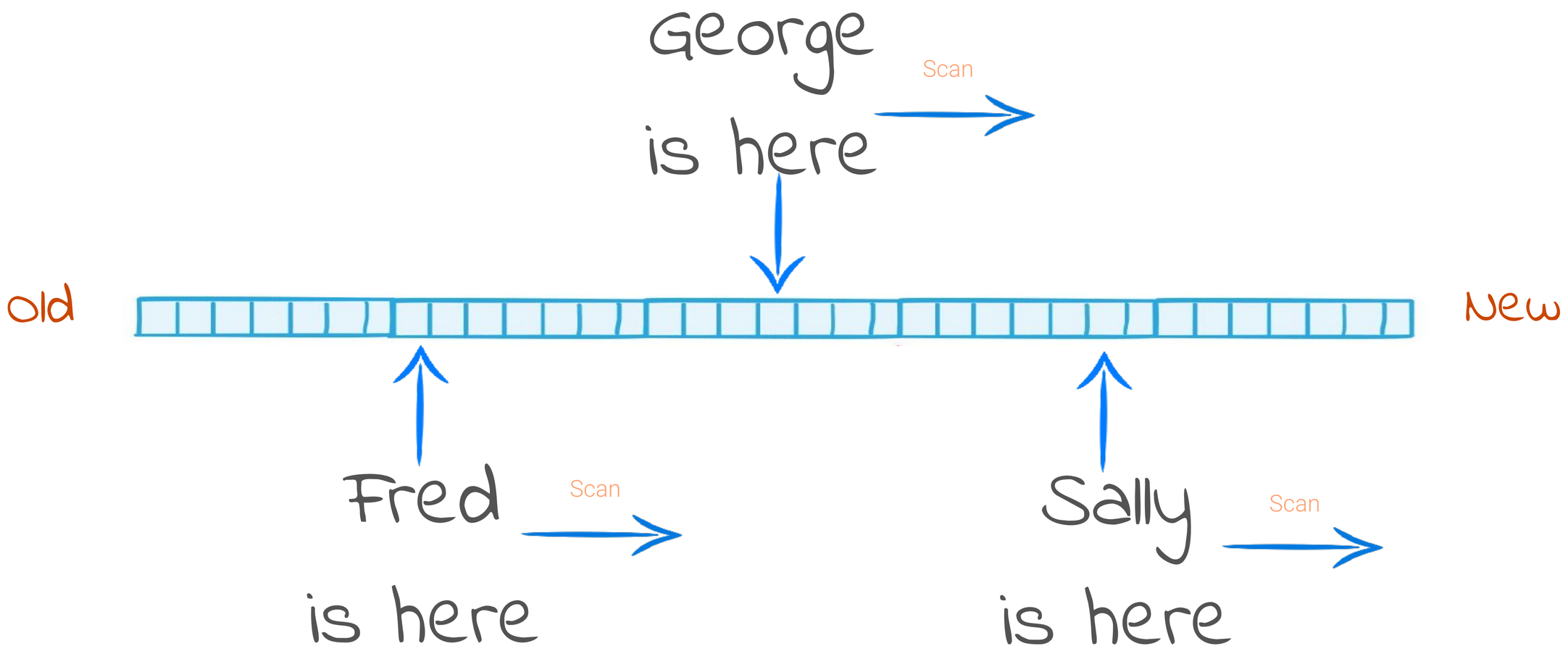
Consumers have a position all of their own



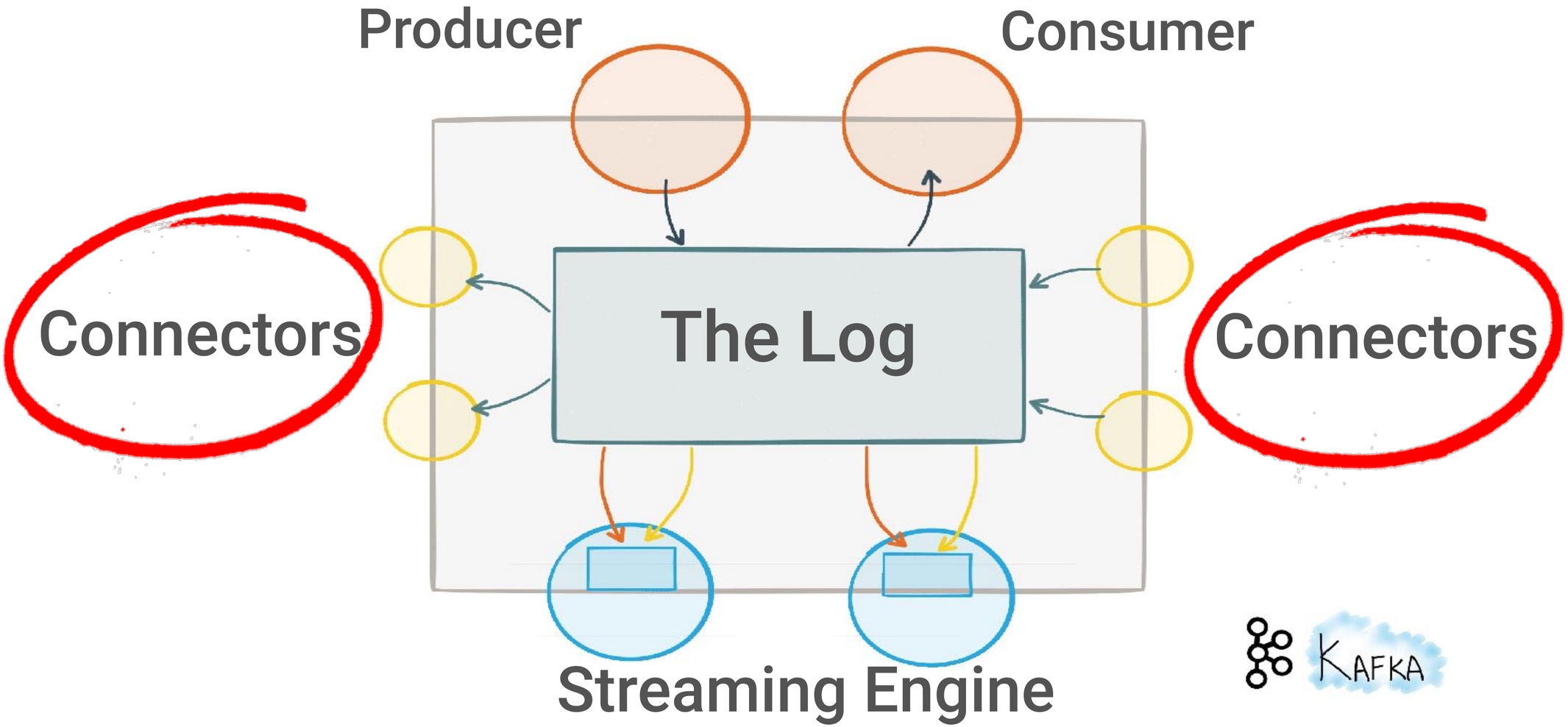
Consumers have a position all of their own



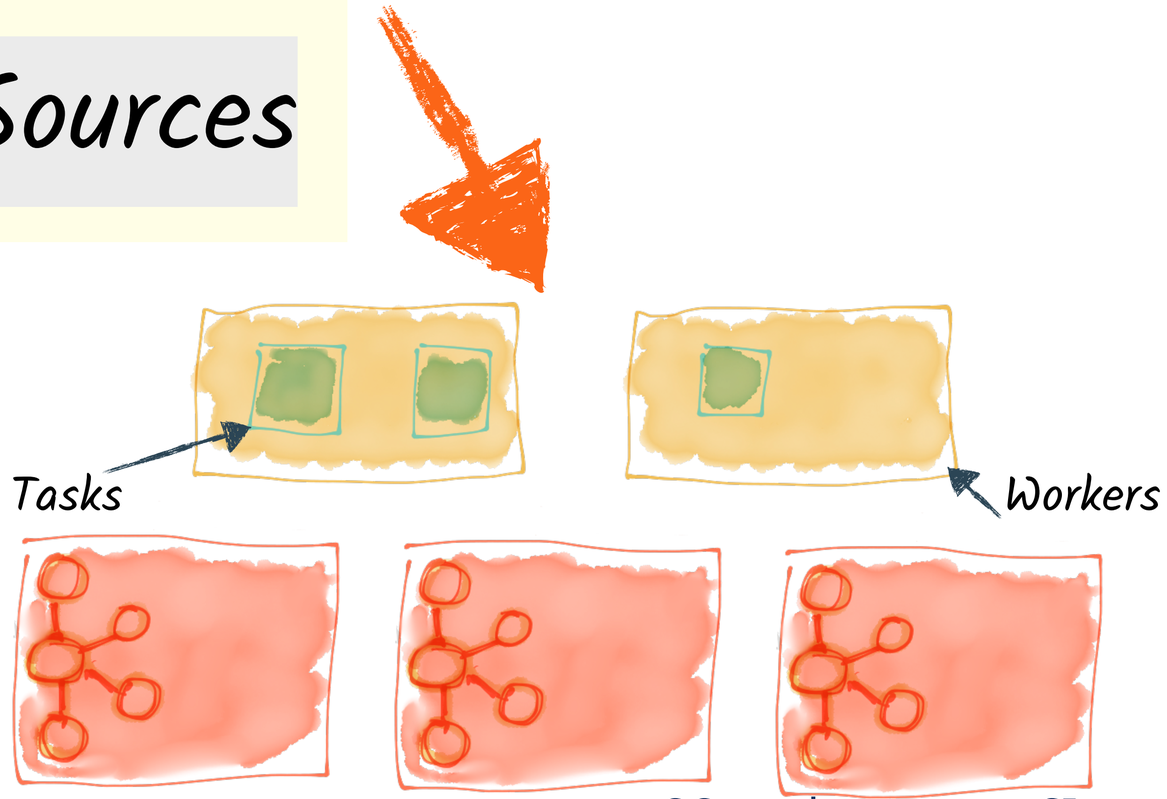
Consumers have a position all of their own



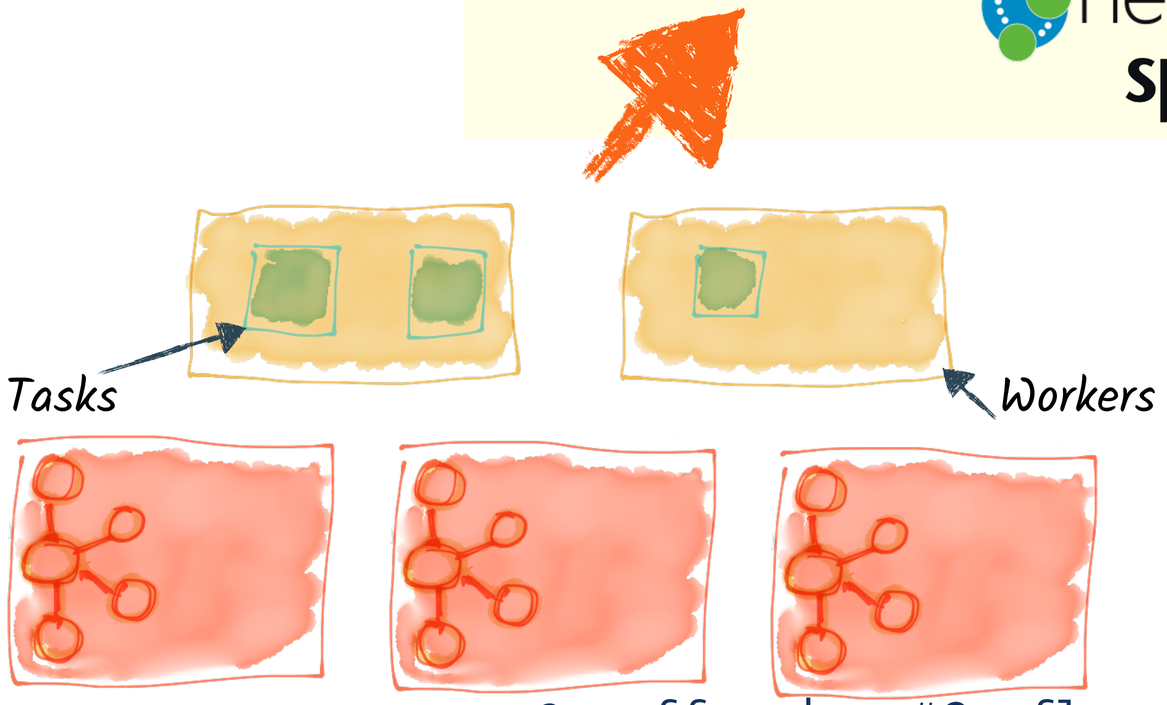
The Connect API



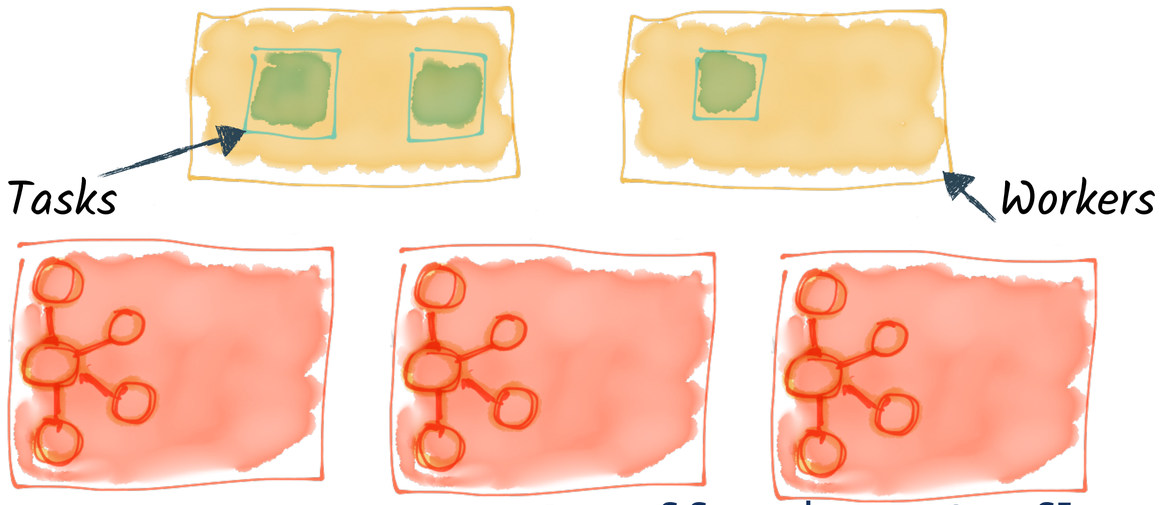
Streaming Integration with Kafka Connect



Streaming Integration with Kafka Connect



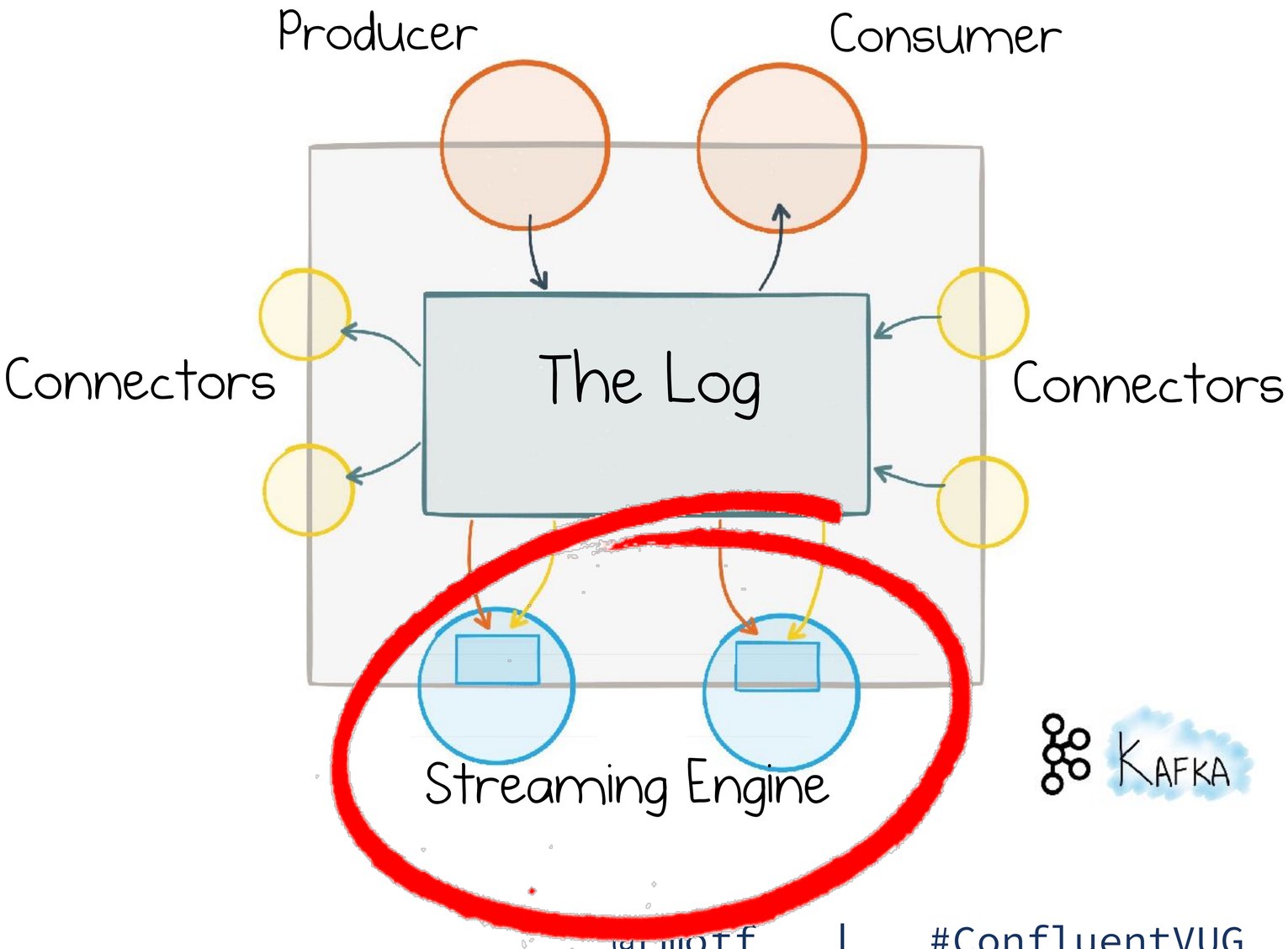
Streaming Integration with Kafka Connect



Kafka Connect

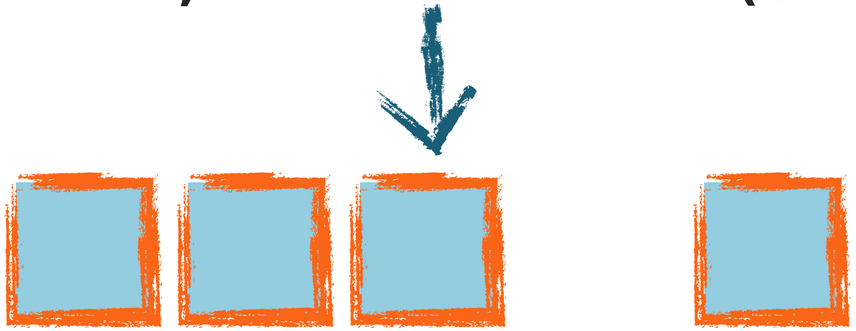
Kafka Brokers

Stream Processing in Kafka



Kafka Streams API

```
final StreamsBuilder builder = new StreamsBuilder()
    .stream("orders", Consumed.with(stringSerde, ordersSerde))
    .filter( (key, order) -> order.getStatus().equals("COMPLETE") )
    .to("complete_orders", Produced.with(stringSerde, ordersSerde));
```



Stream Processing with ksqlDB

```
CREATE STREAM completedOrders AS
SELECT *
FROM orders
WHERE status='COMPLETE';
```

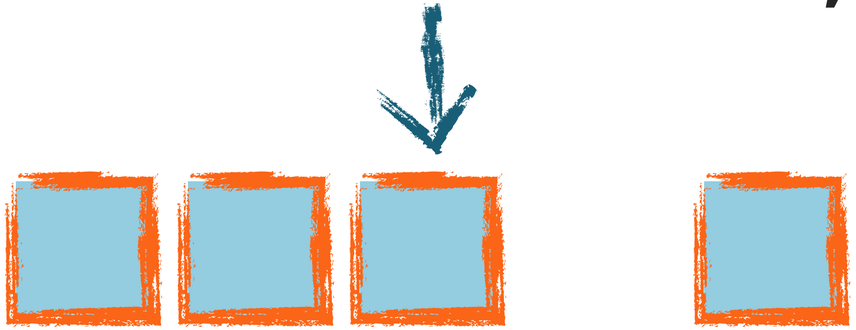
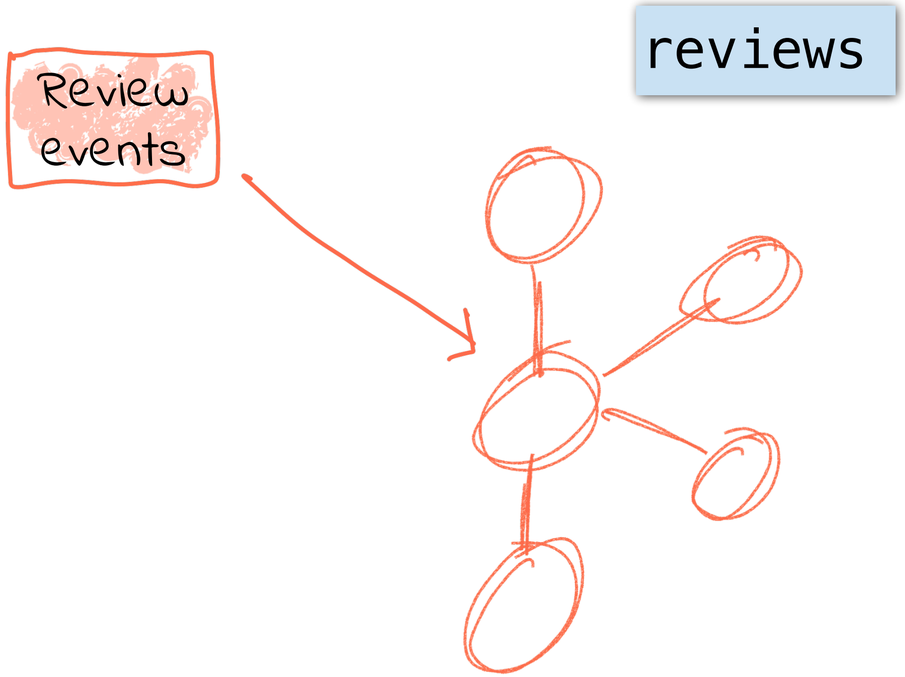
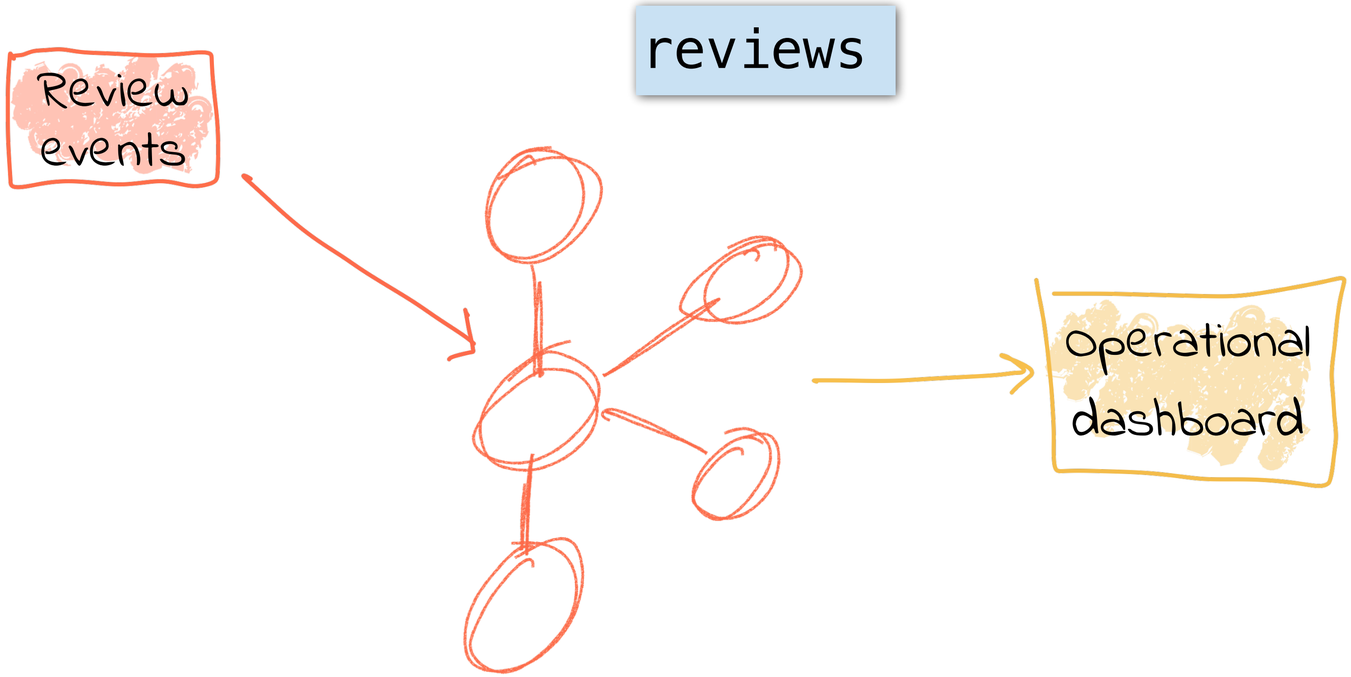


Photo by Ash from Modern Affiatus on Unsplash

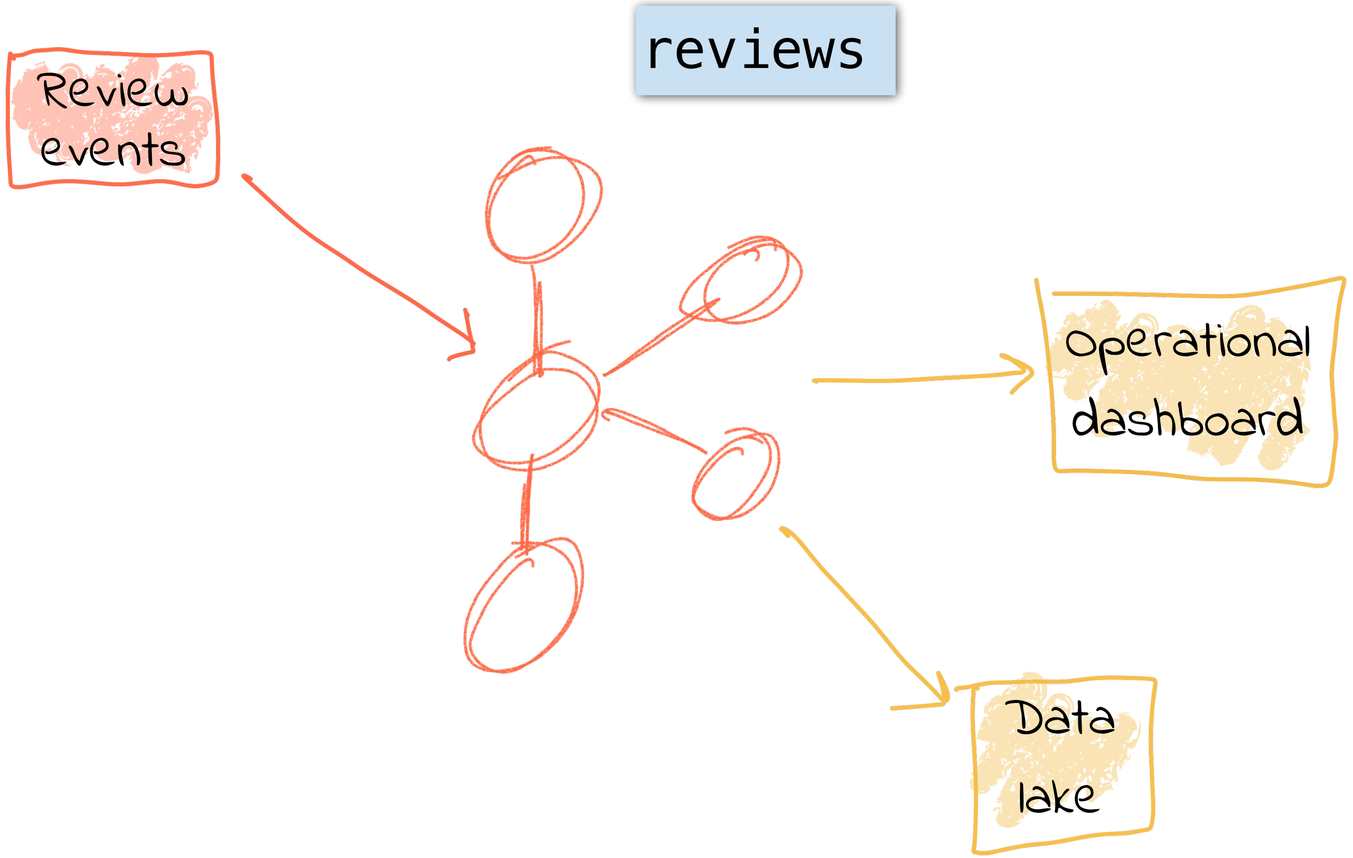
Events in Action



Events in Action

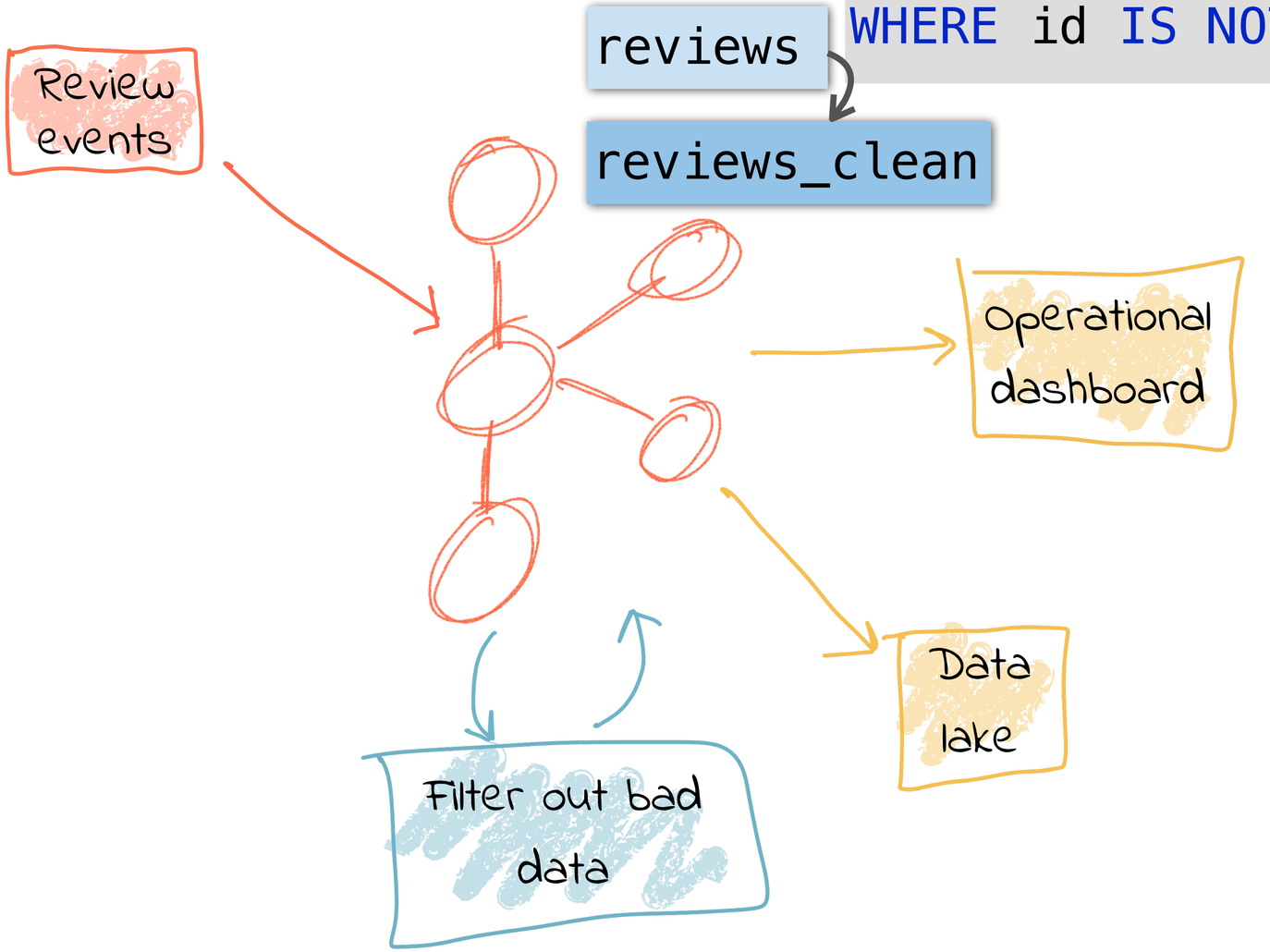


Events in Action

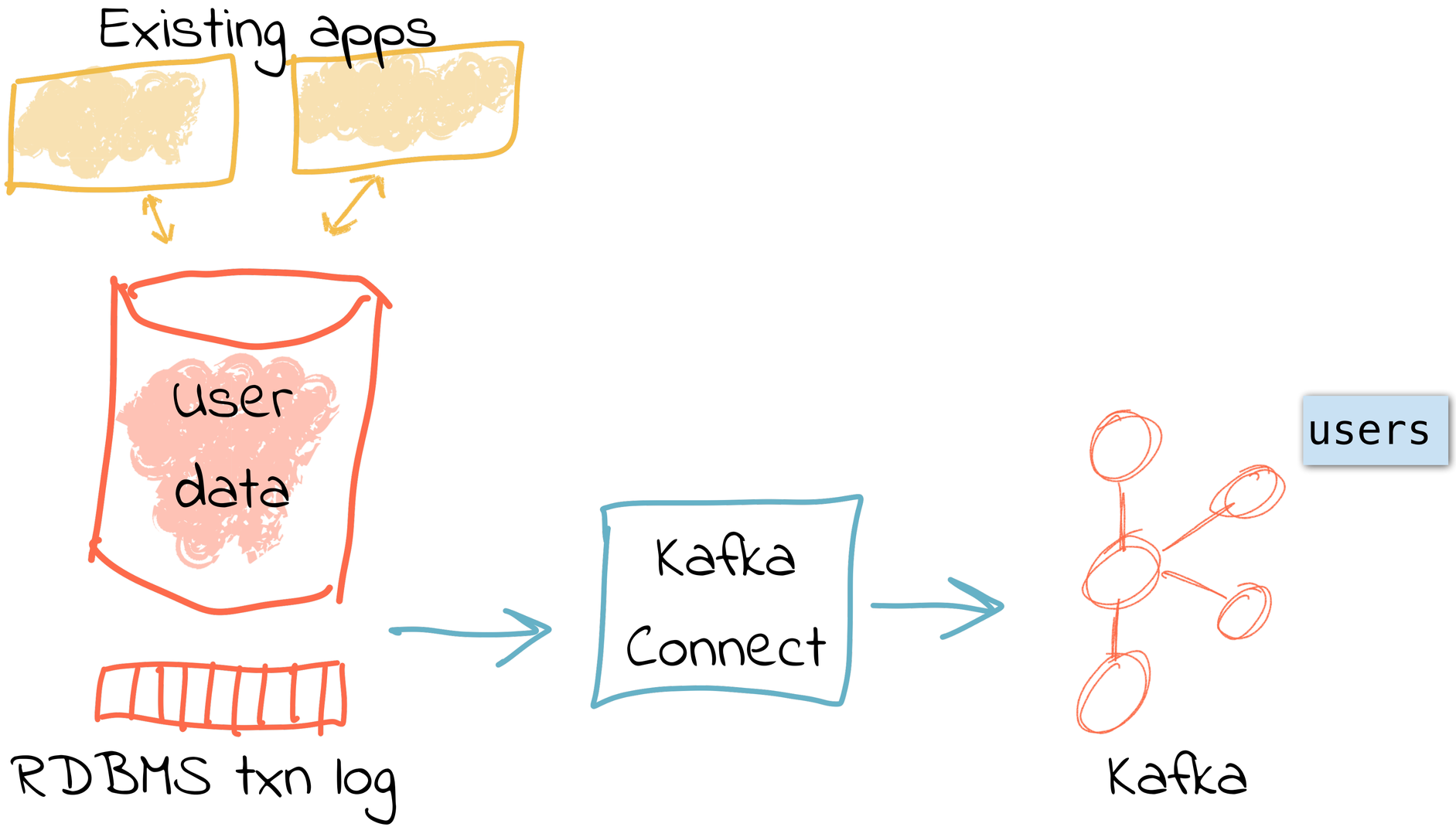


Events in Action

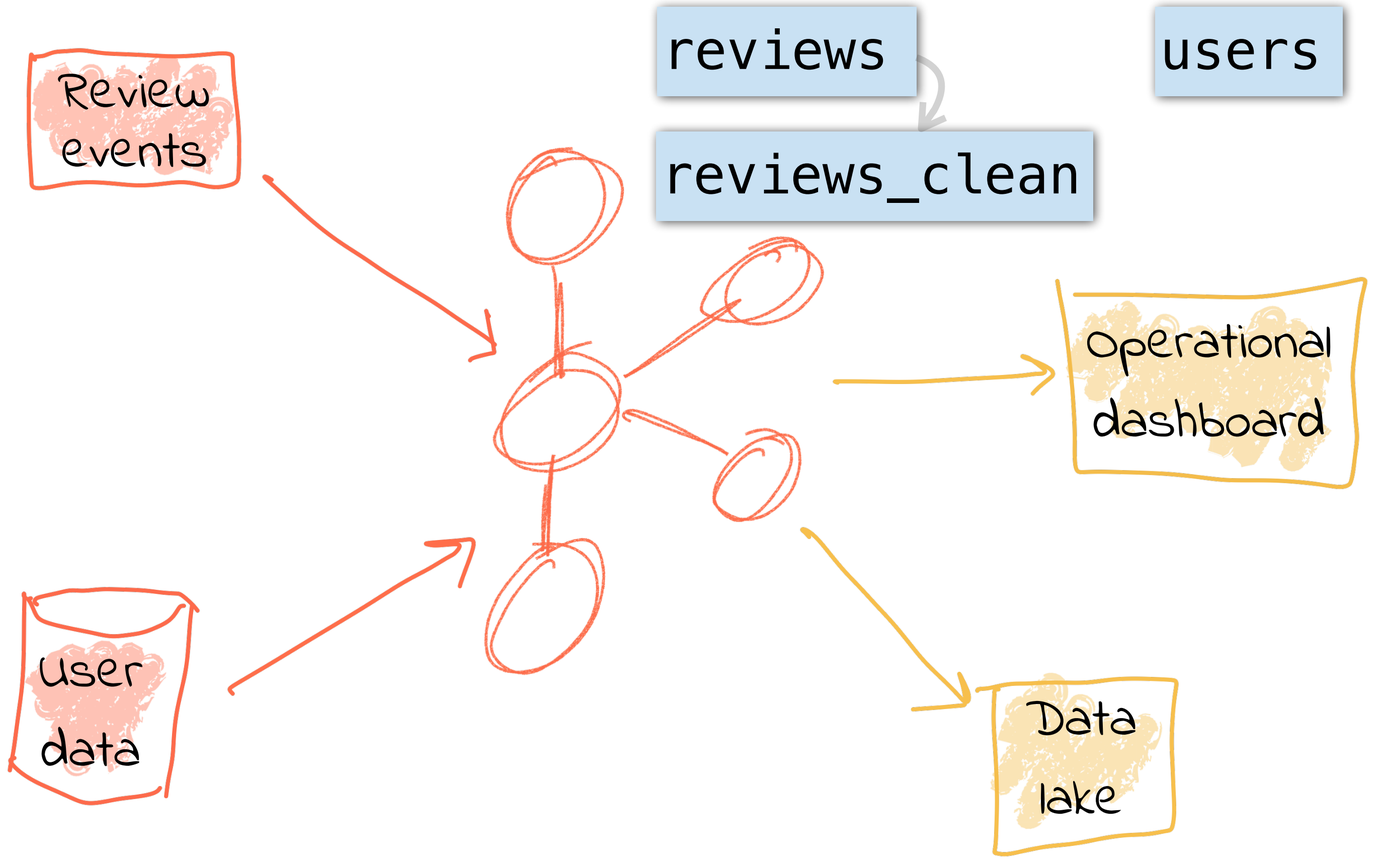
```
CREATE STREAM reviews_clean AS
SELECT * FROM reviews
WHERE id IS NOT NULL;
```



Events in Action

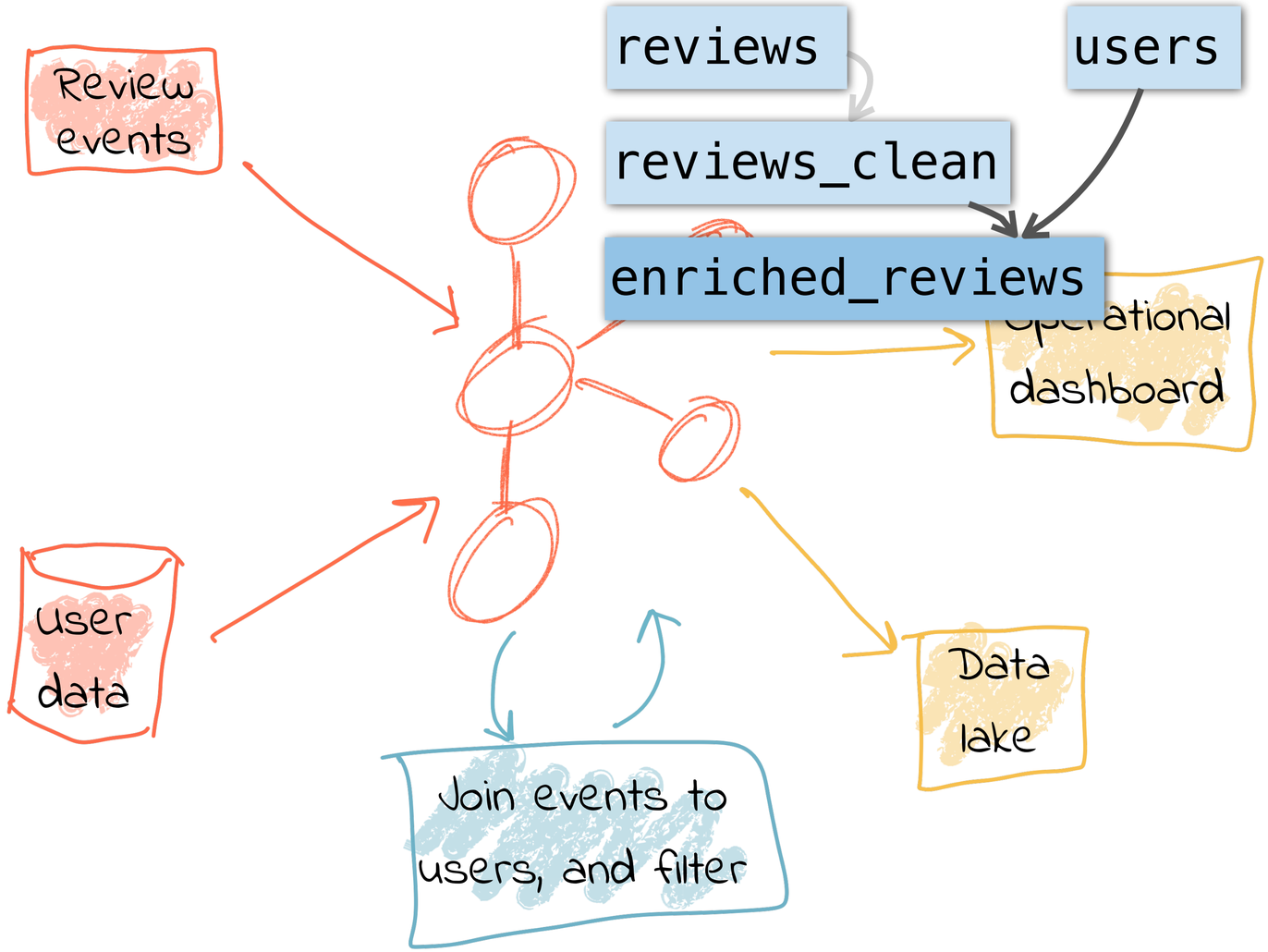


Events in Action

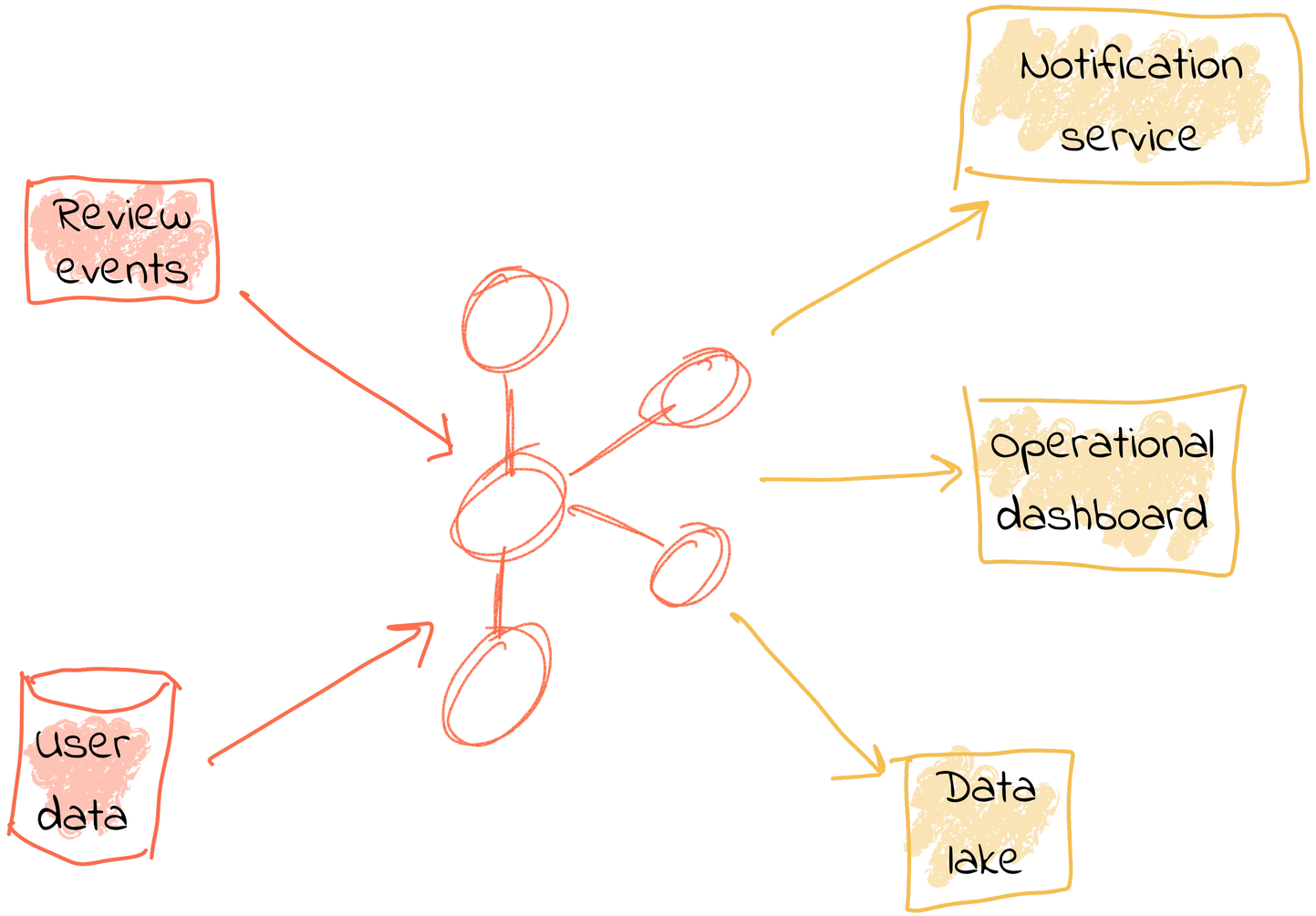


Events in Action

```
CREATE STREAM enriched_reviews AS
SELECT * FROM reviews_clean r
         INNER JOIN users u
         ON r.userid=u.userid;
```

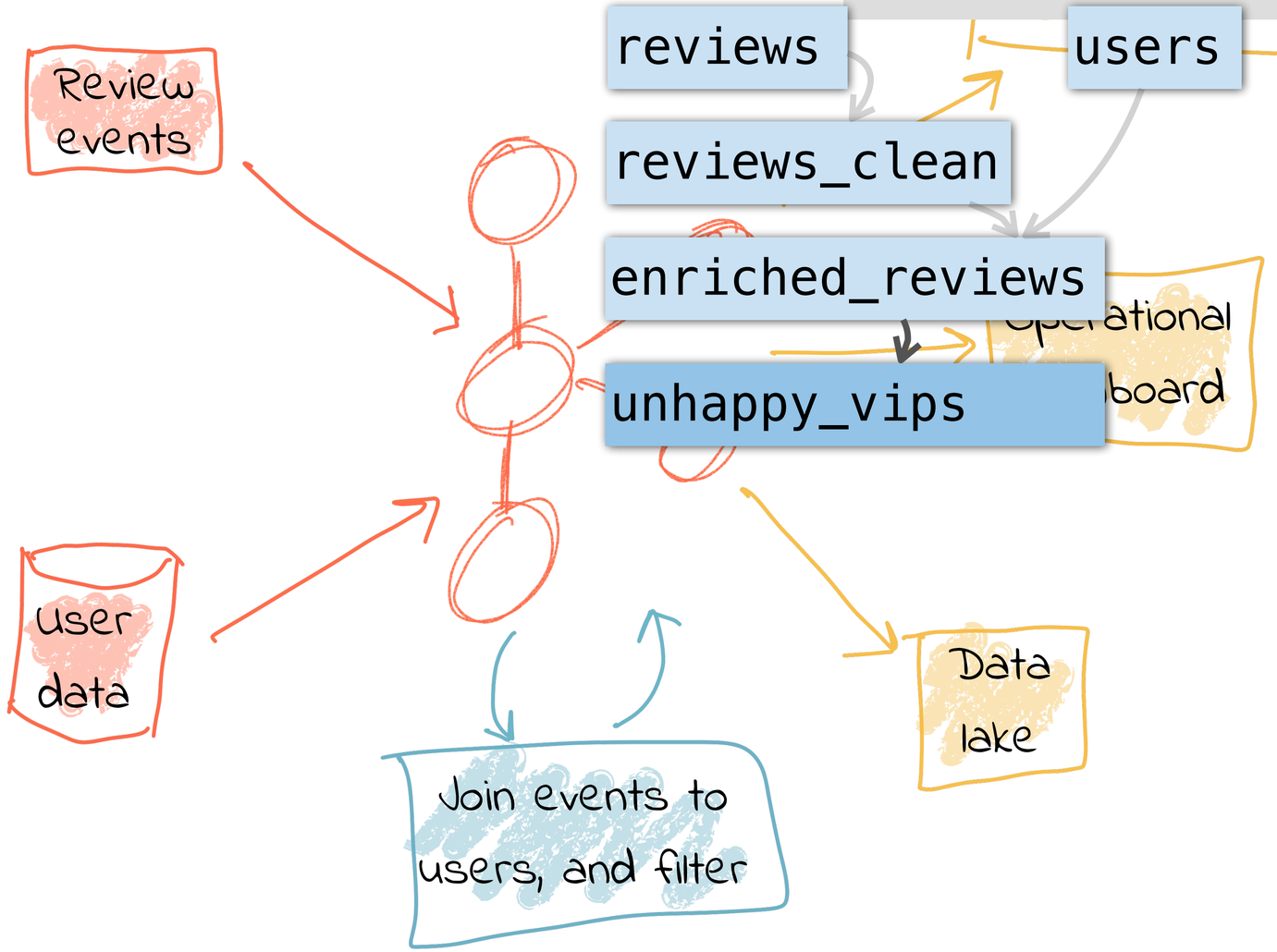


Events in Action



Events in Action

```
CREATE STREAM unhappy_vips AS
SELECT * FROM enriched_reviews
WHERE rating < 3
AND   status = 'Platinum';
```



The Power of an Event-Driven Architecture

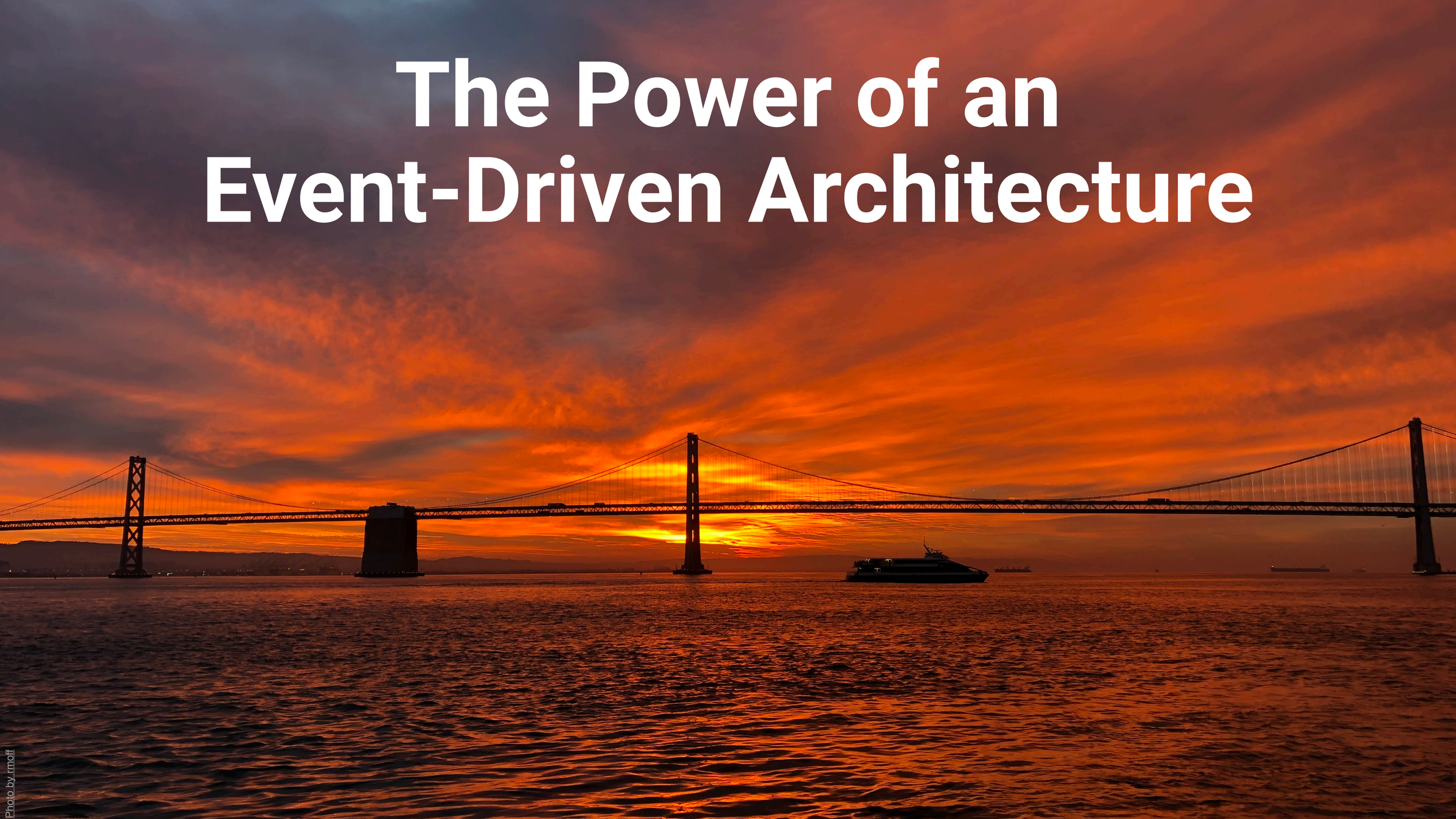
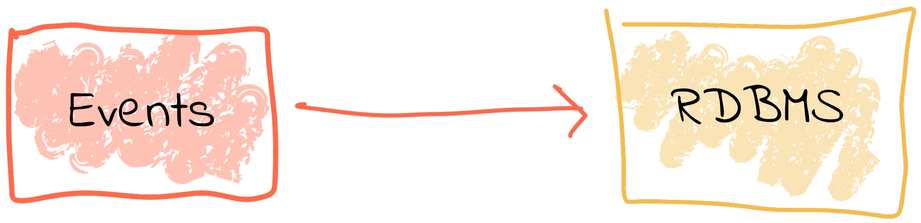
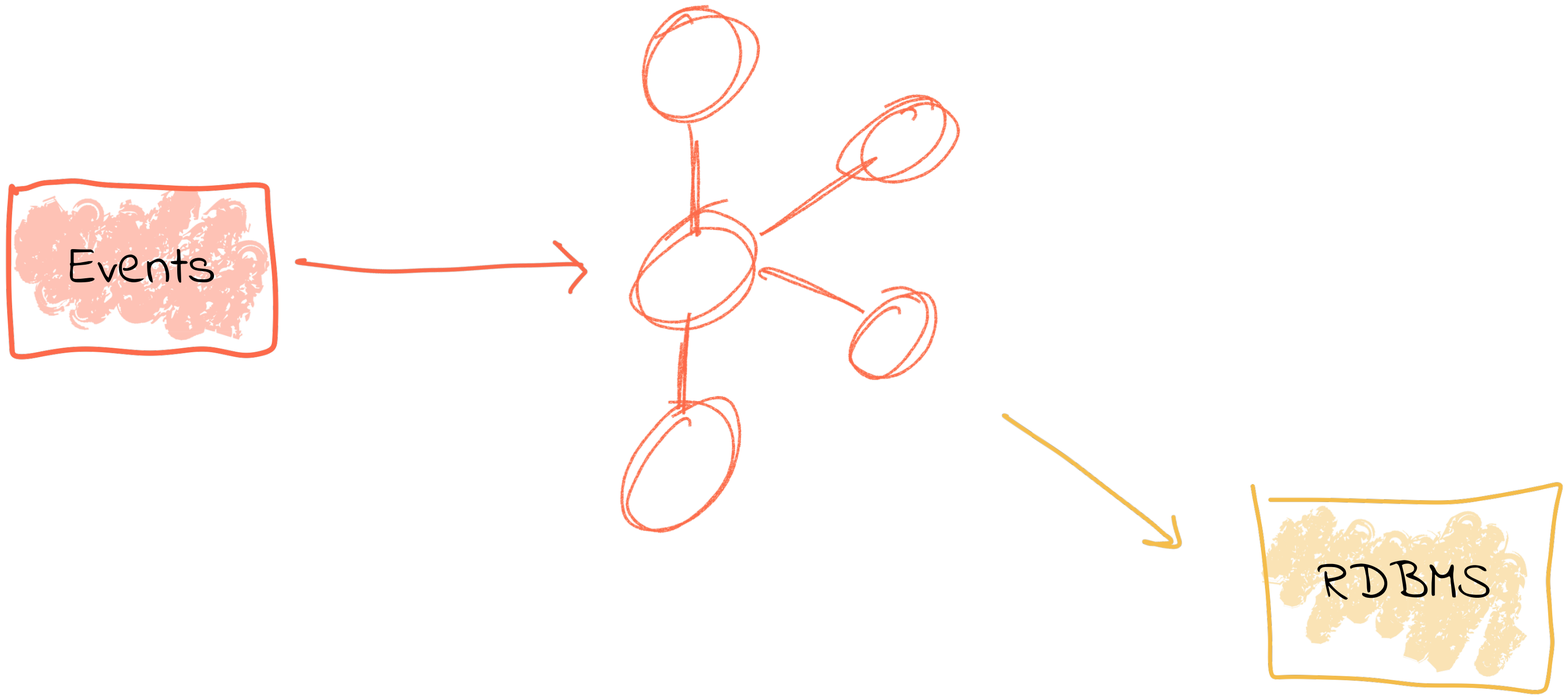


Photo by rmoff

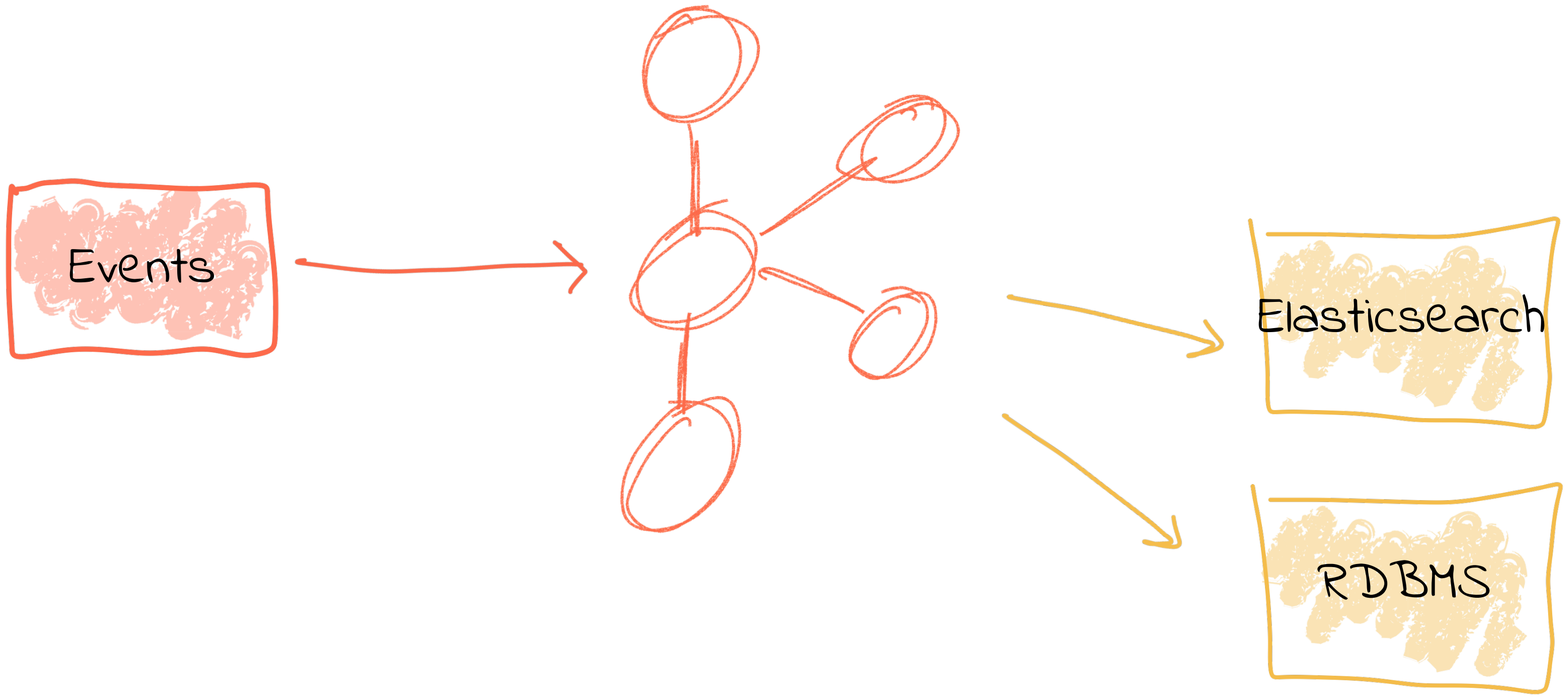
Not Everything is a Nail



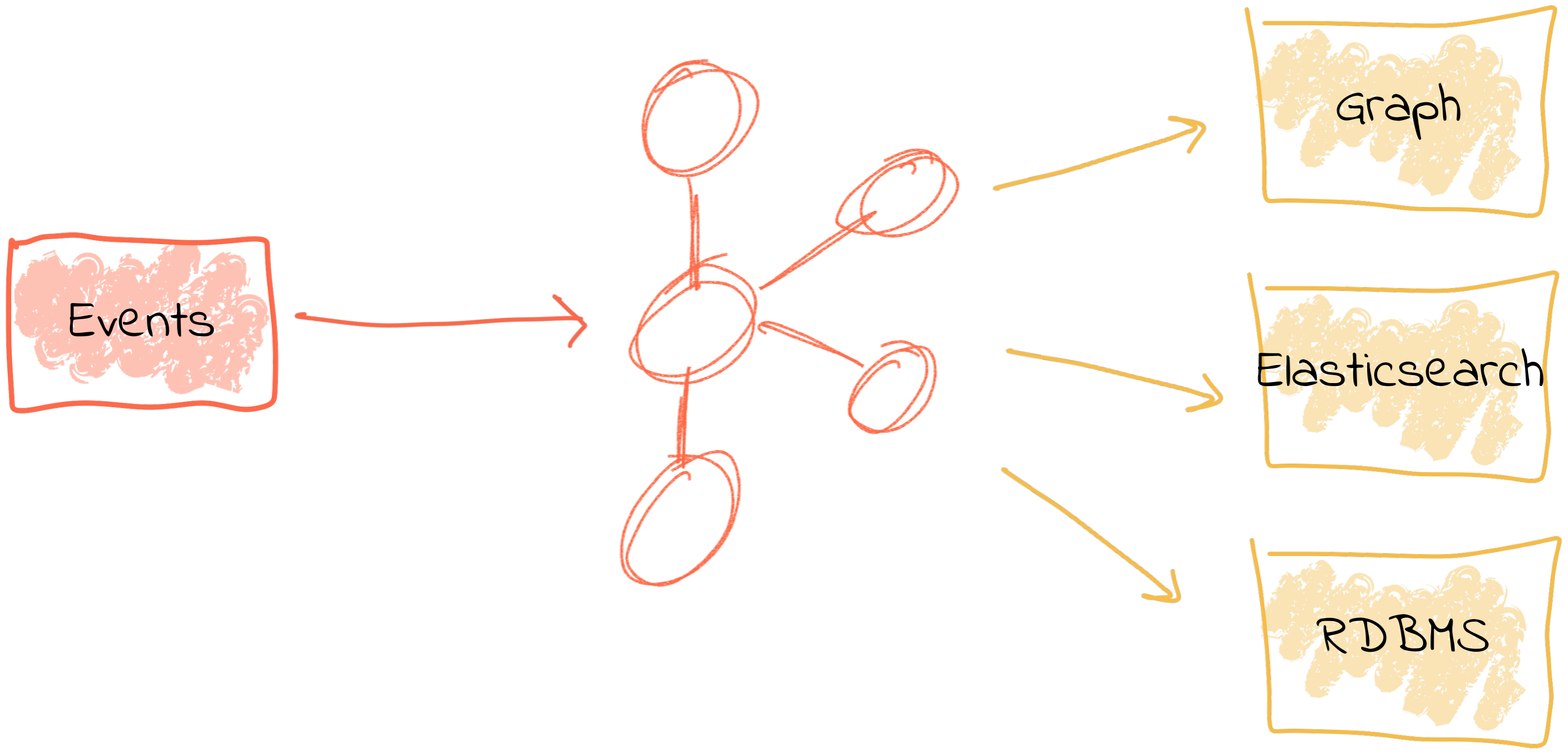
Not Everything is a Nail



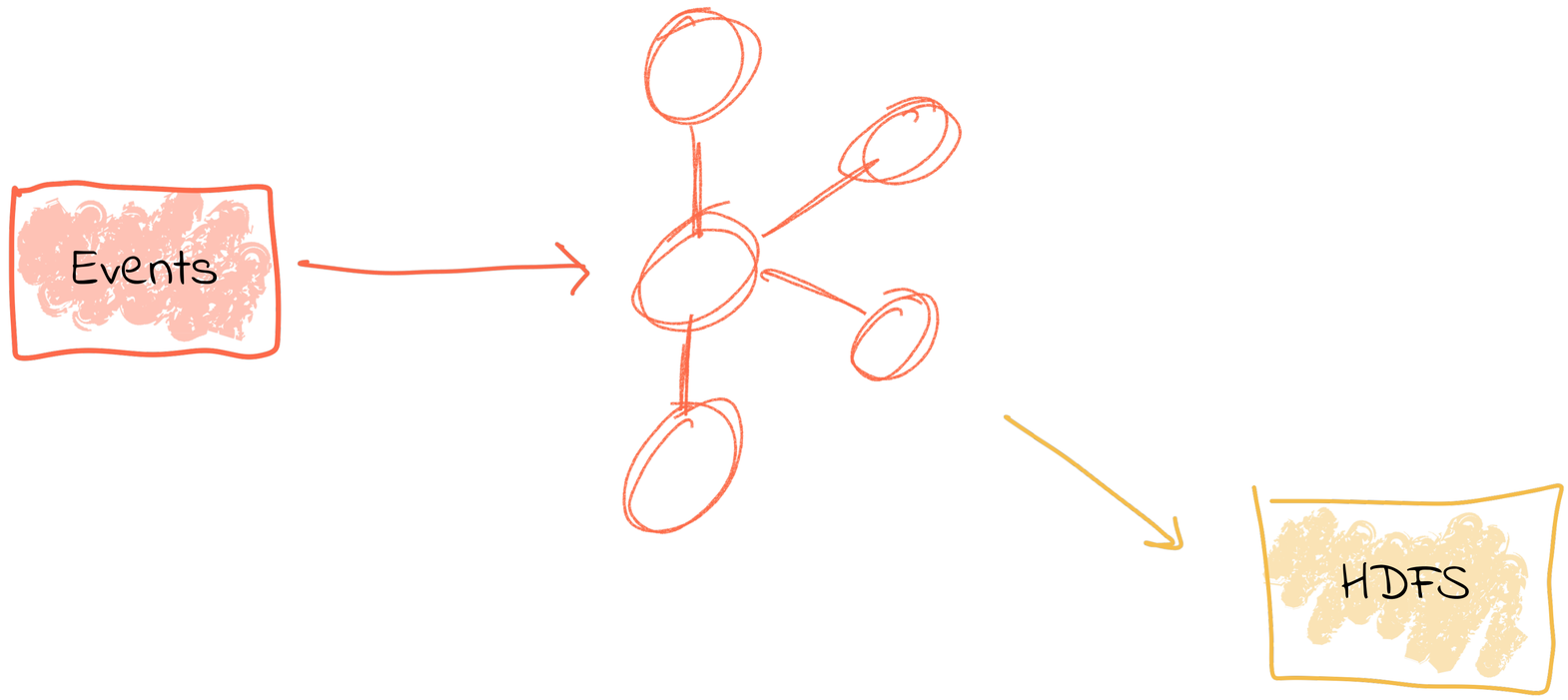
Not Everything is a Nail



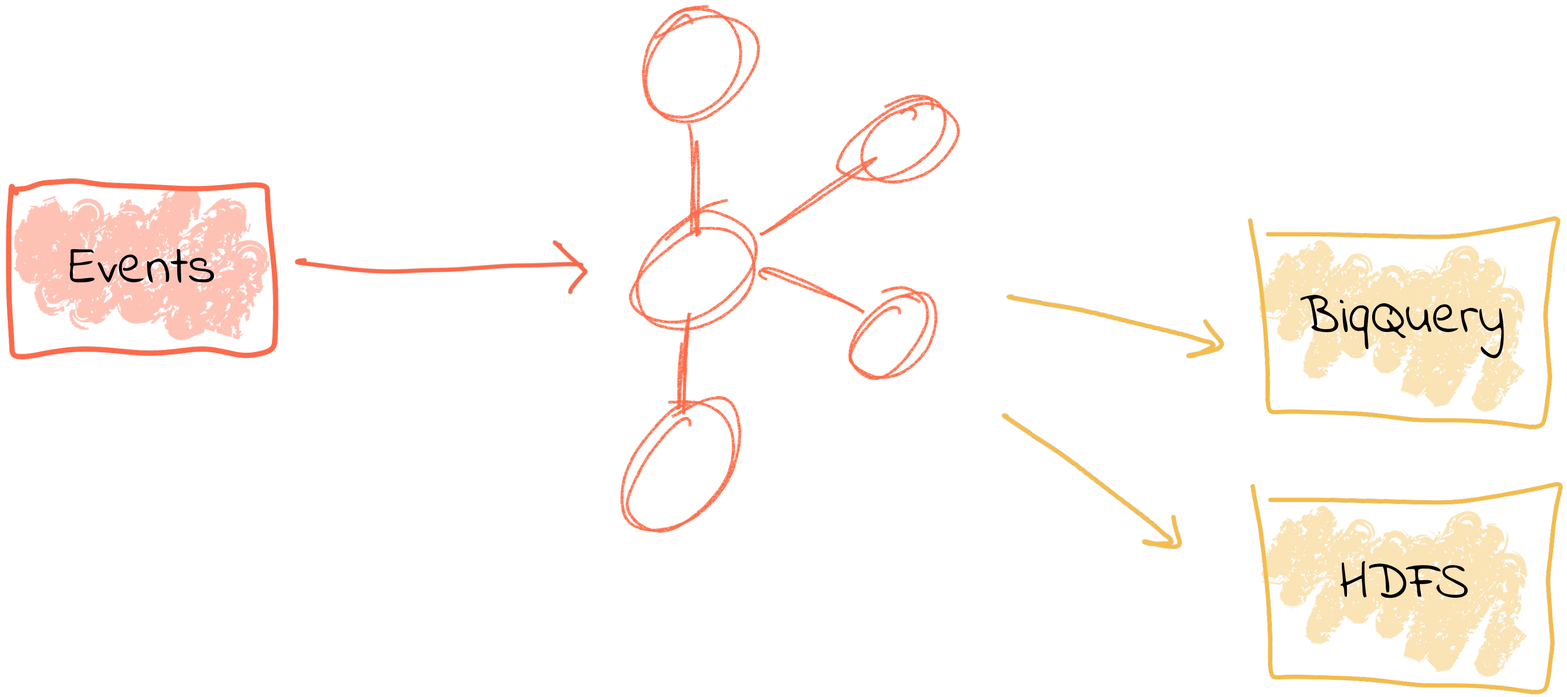
Not Everything is a Nail



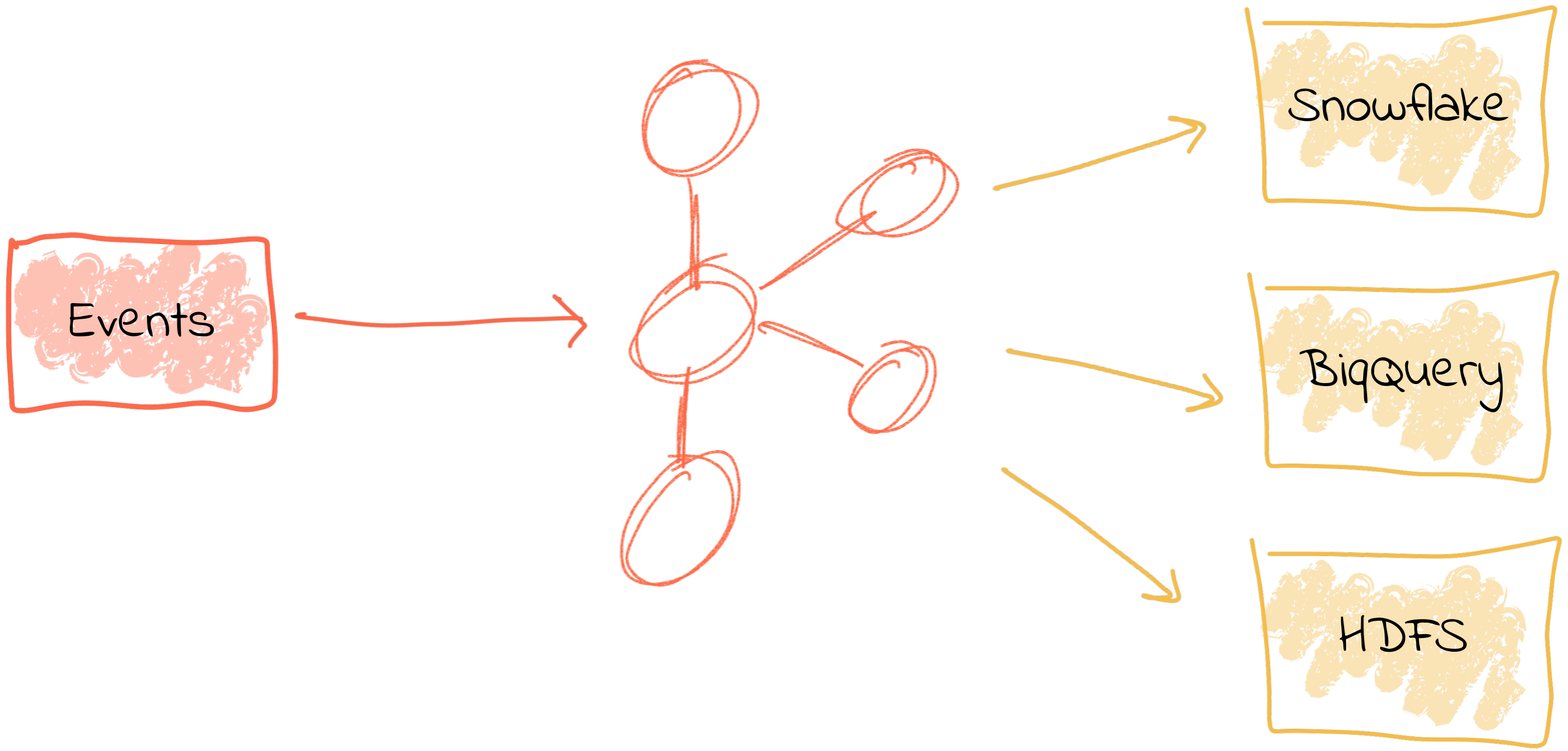
Side-by-Side Tech Evaluation



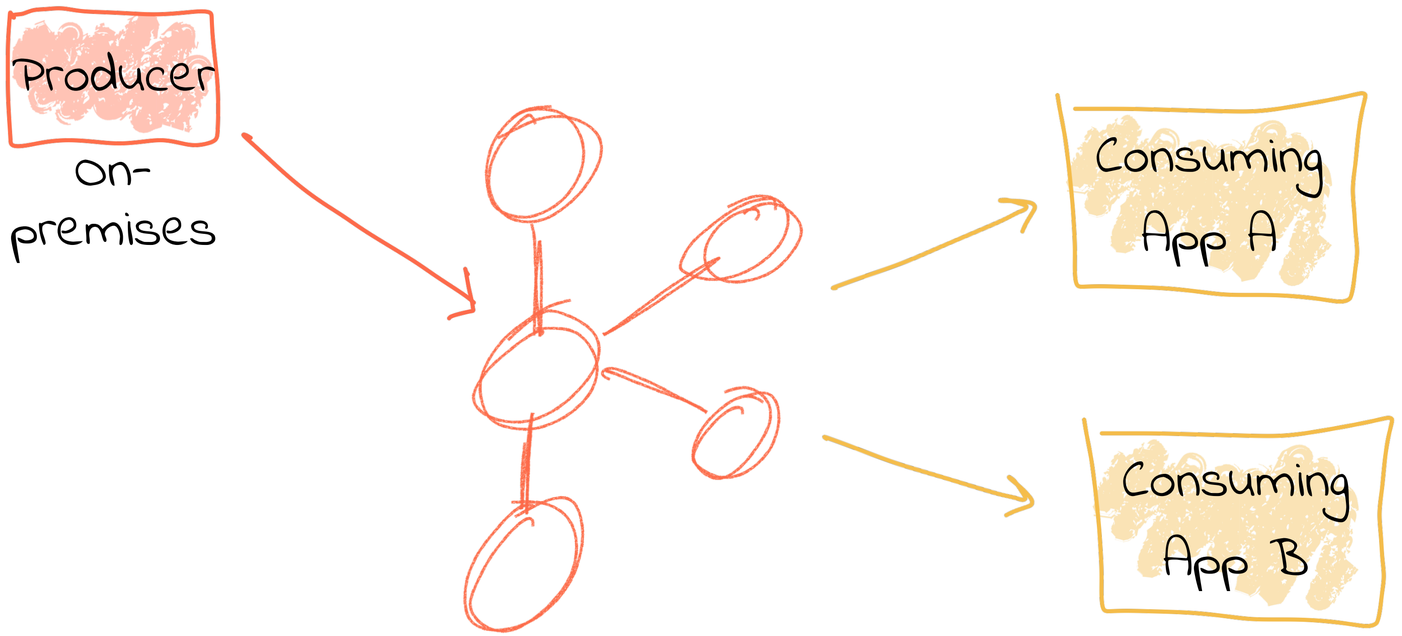
Side-by-Side Tech Evaluation



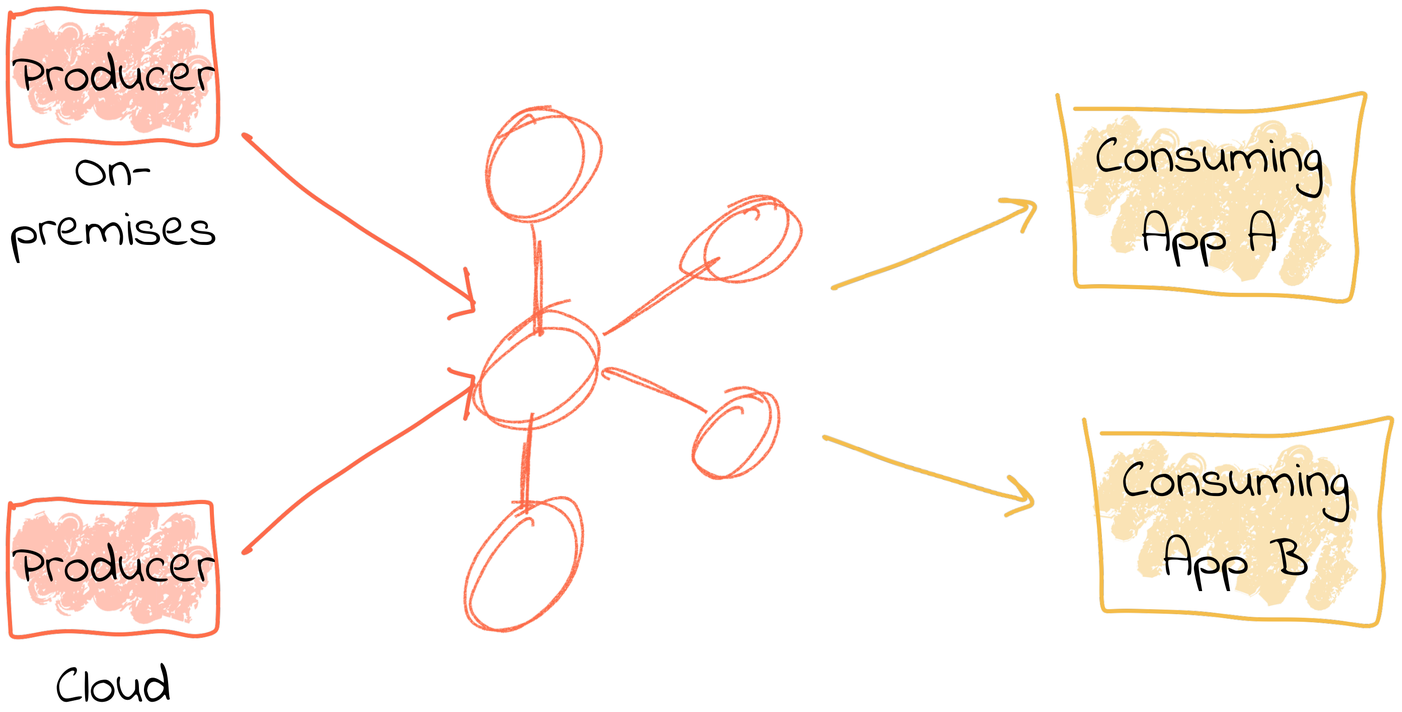
Side-by-Side Tech Evaluation



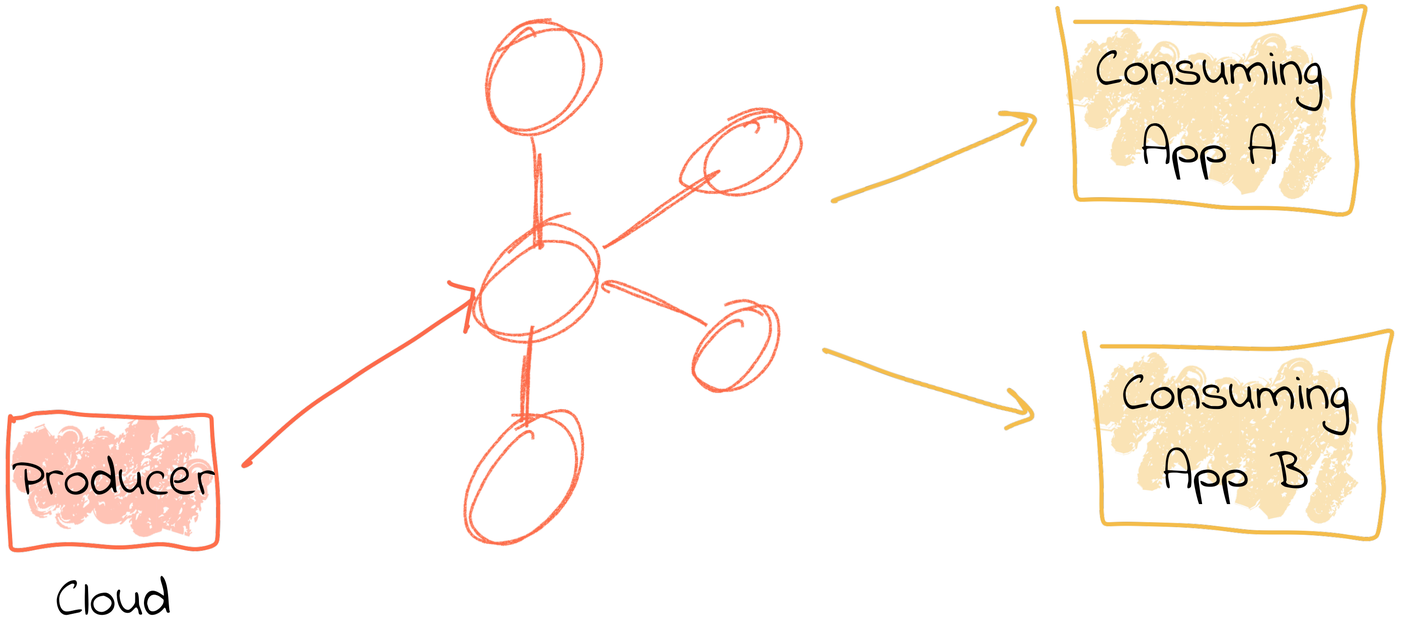
Evolve Data Sources



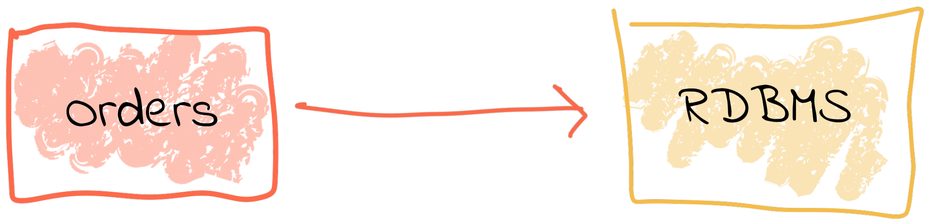
Evolve Data Sources



Evolve Data Sources

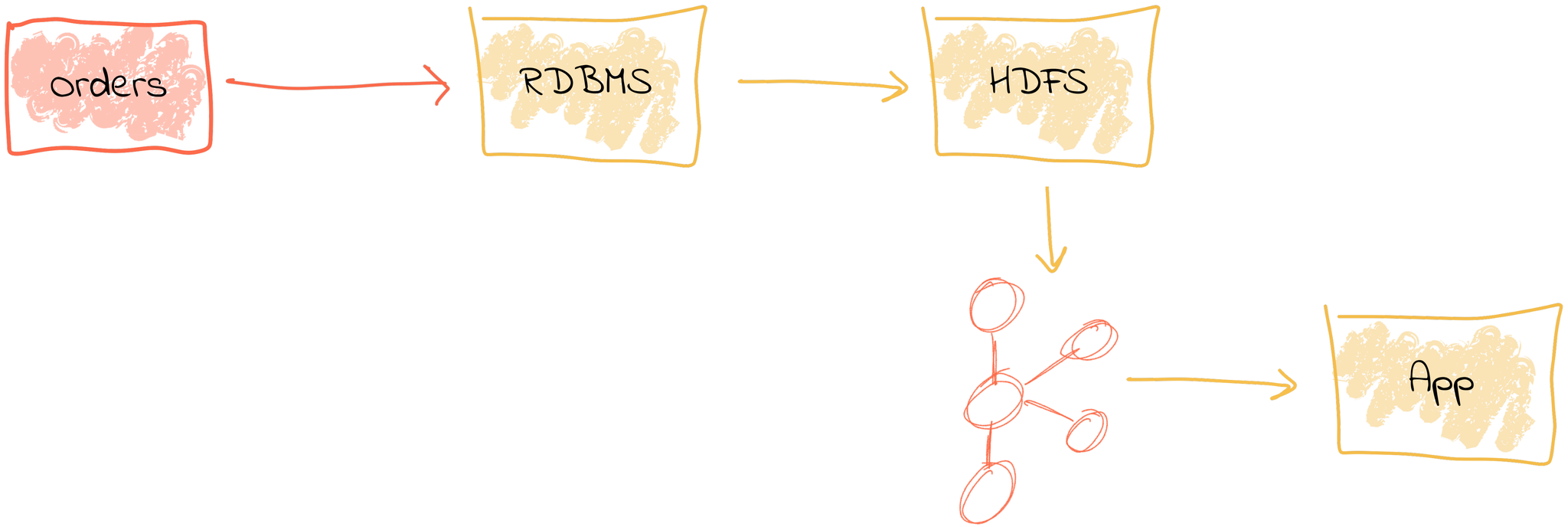


Tight Coupling != Flexible

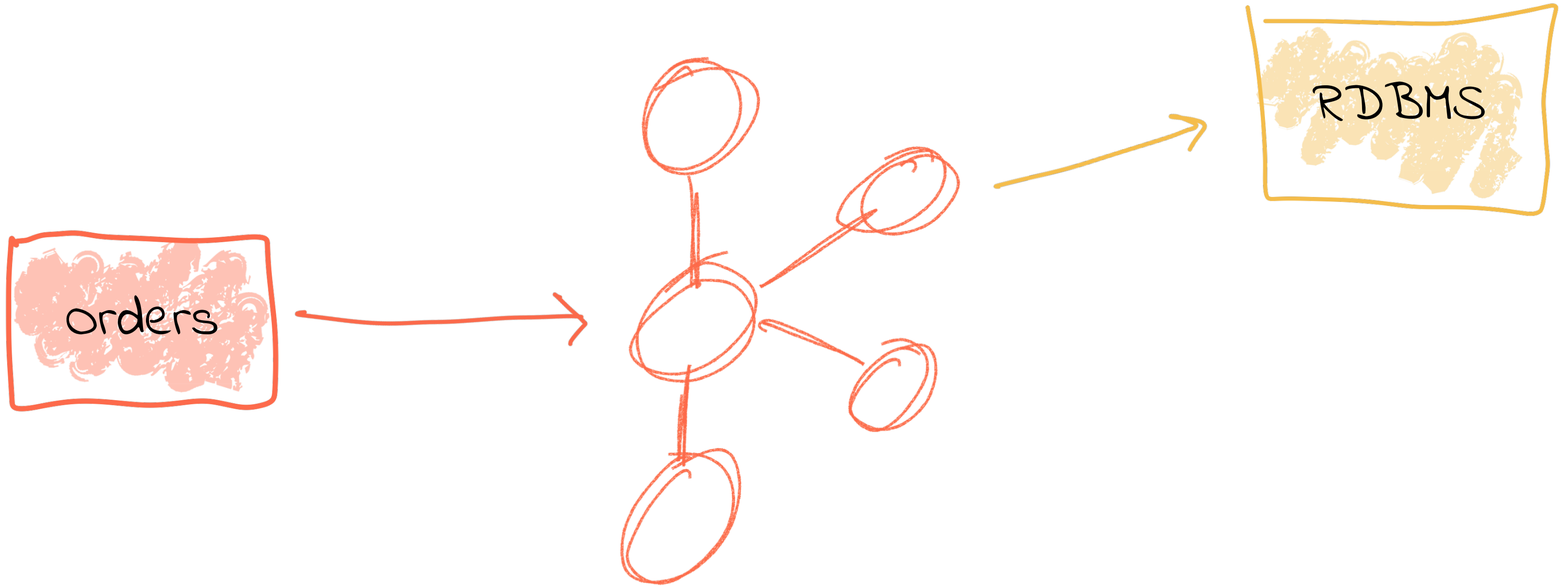


Tight Coupling != Flexible

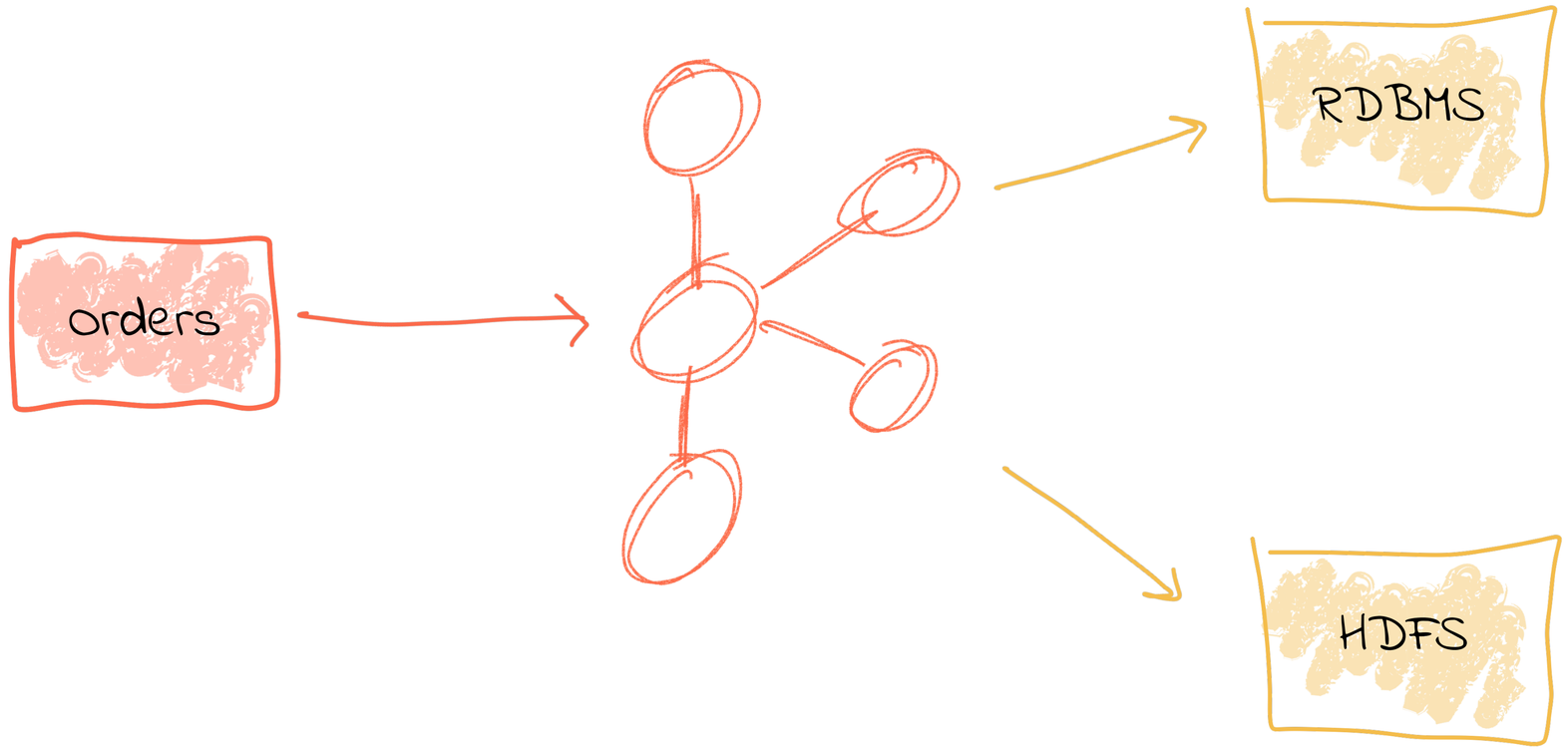
Tight Coupling != Flexible



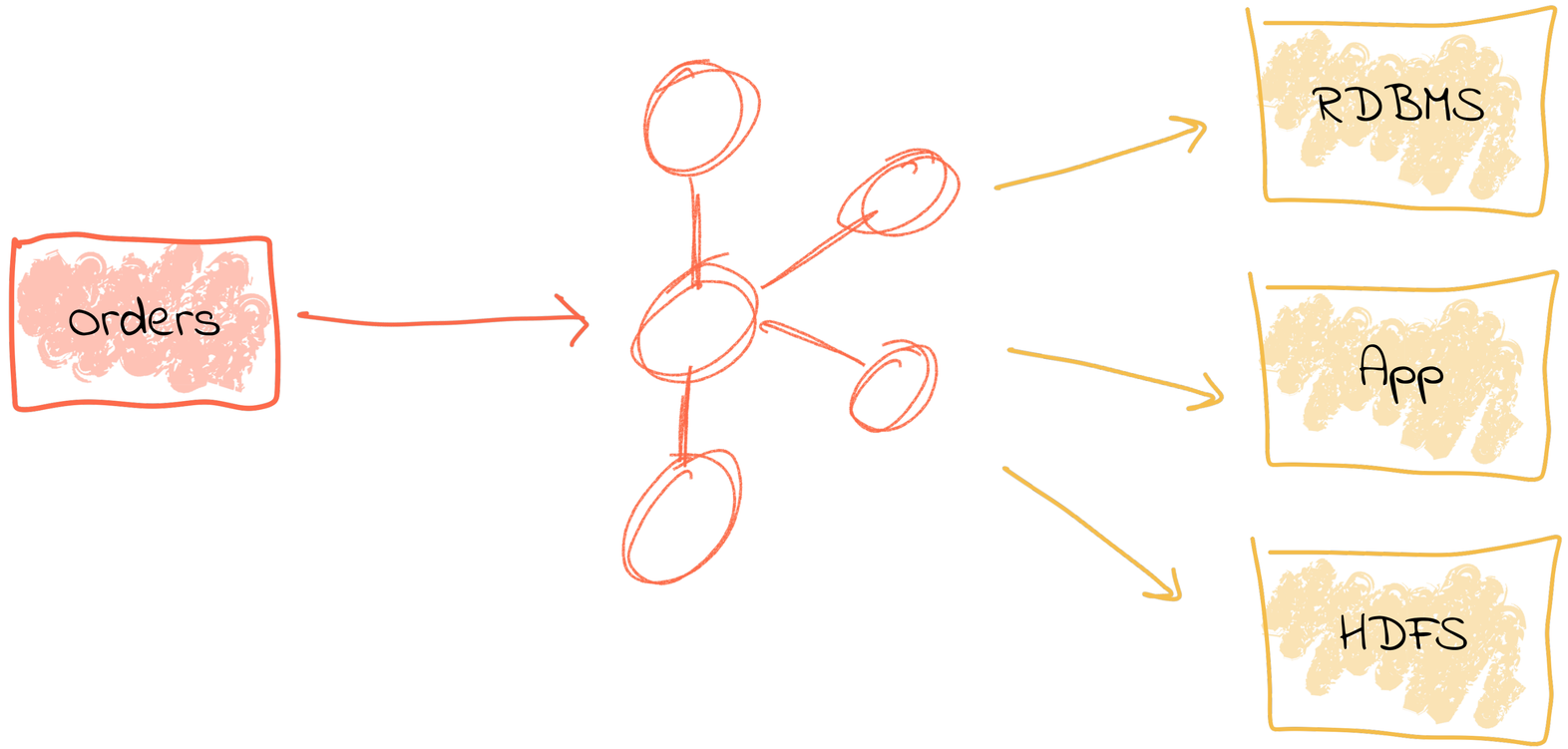
Loose Coupling == Freedom to Evolve



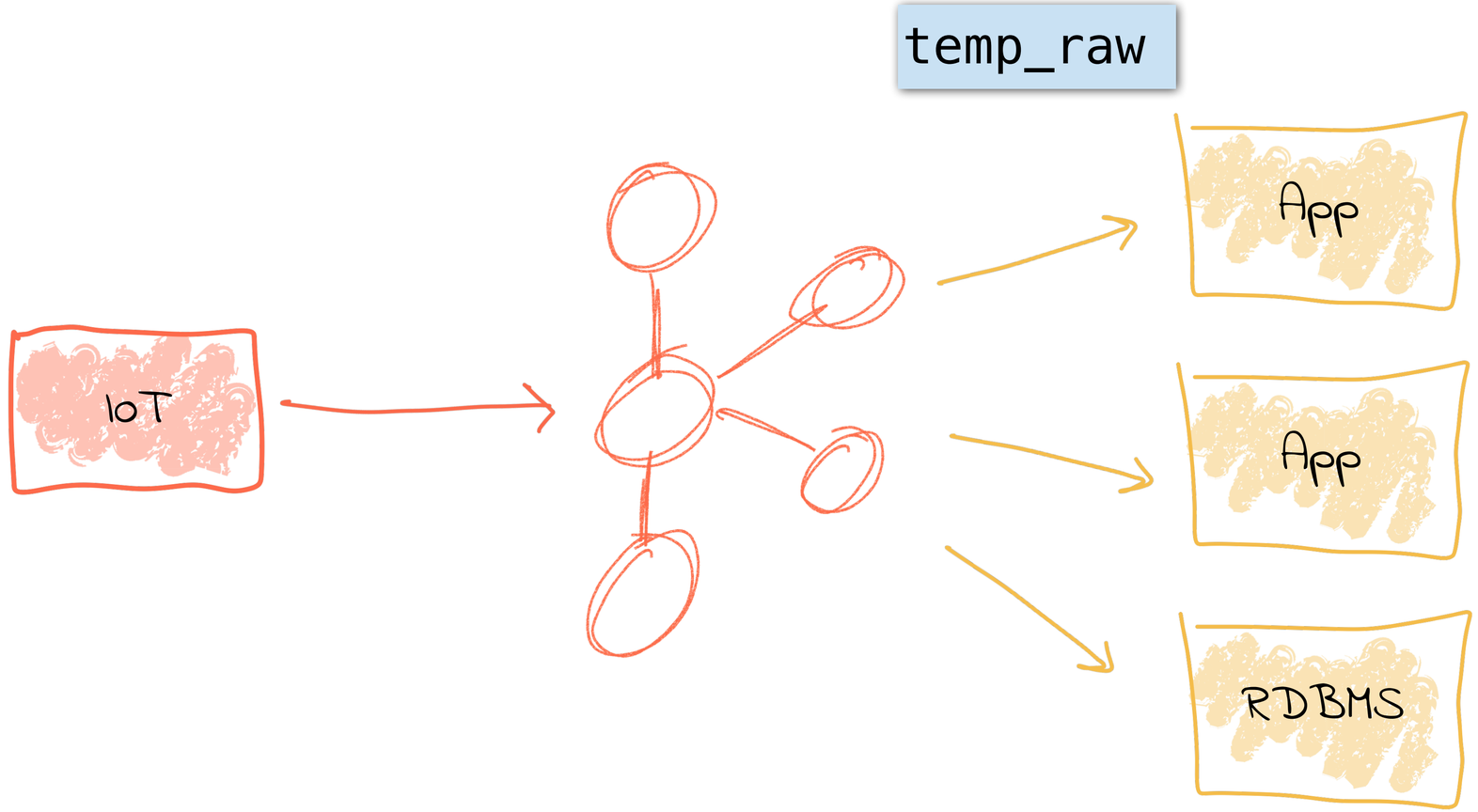
Loose Coupling == Freedom to Evolve



Loose Coupling == Freedom to Evolve

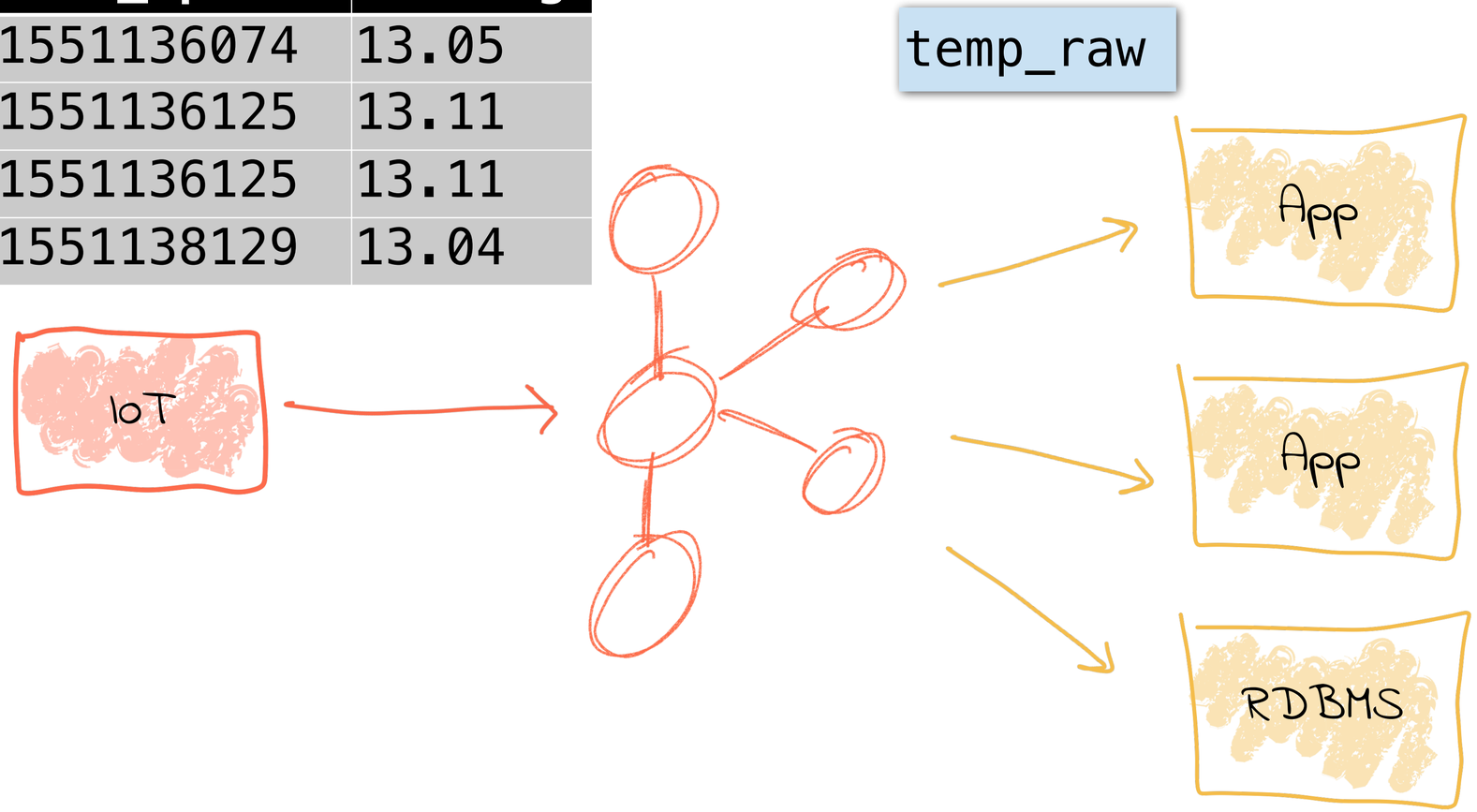


Transform Once, Use Many: Data Cleansing



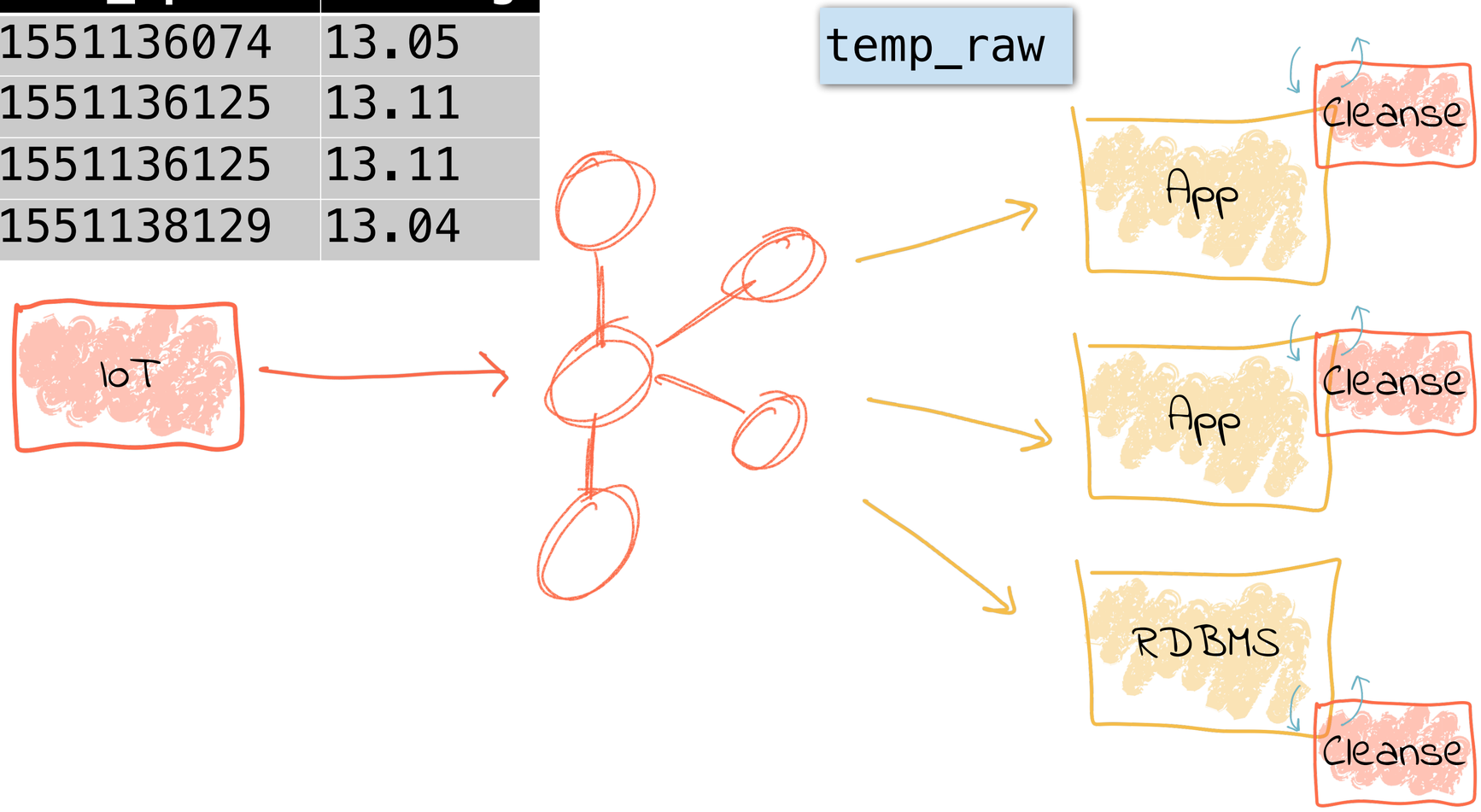
Transform Once, Use Many: Data Cleansing

sensor_id	time_epoch	reading
42	1551136074	13.05
42	1551136125	13.11
	1551136125	13.11
42	1551138129	13.04



Transform Once, Use Many: Data Cleansing

sensor_id	time_epoch	reading
42	1551136074	13.05
42	1551136125	13.11
	1551136125	13.11
42	1551138129	13.04

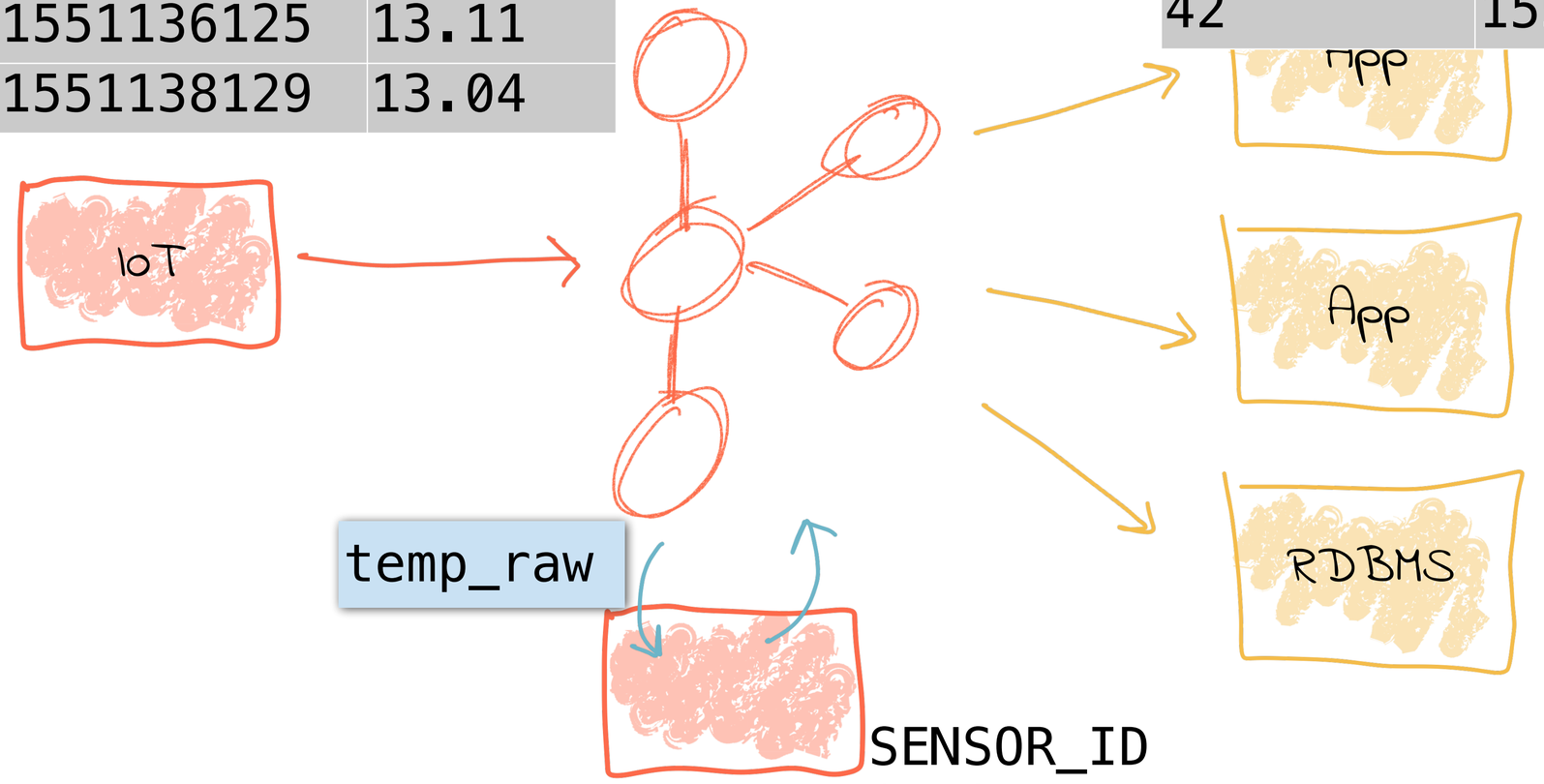


Transform Once, Use Many: Data Cleansing

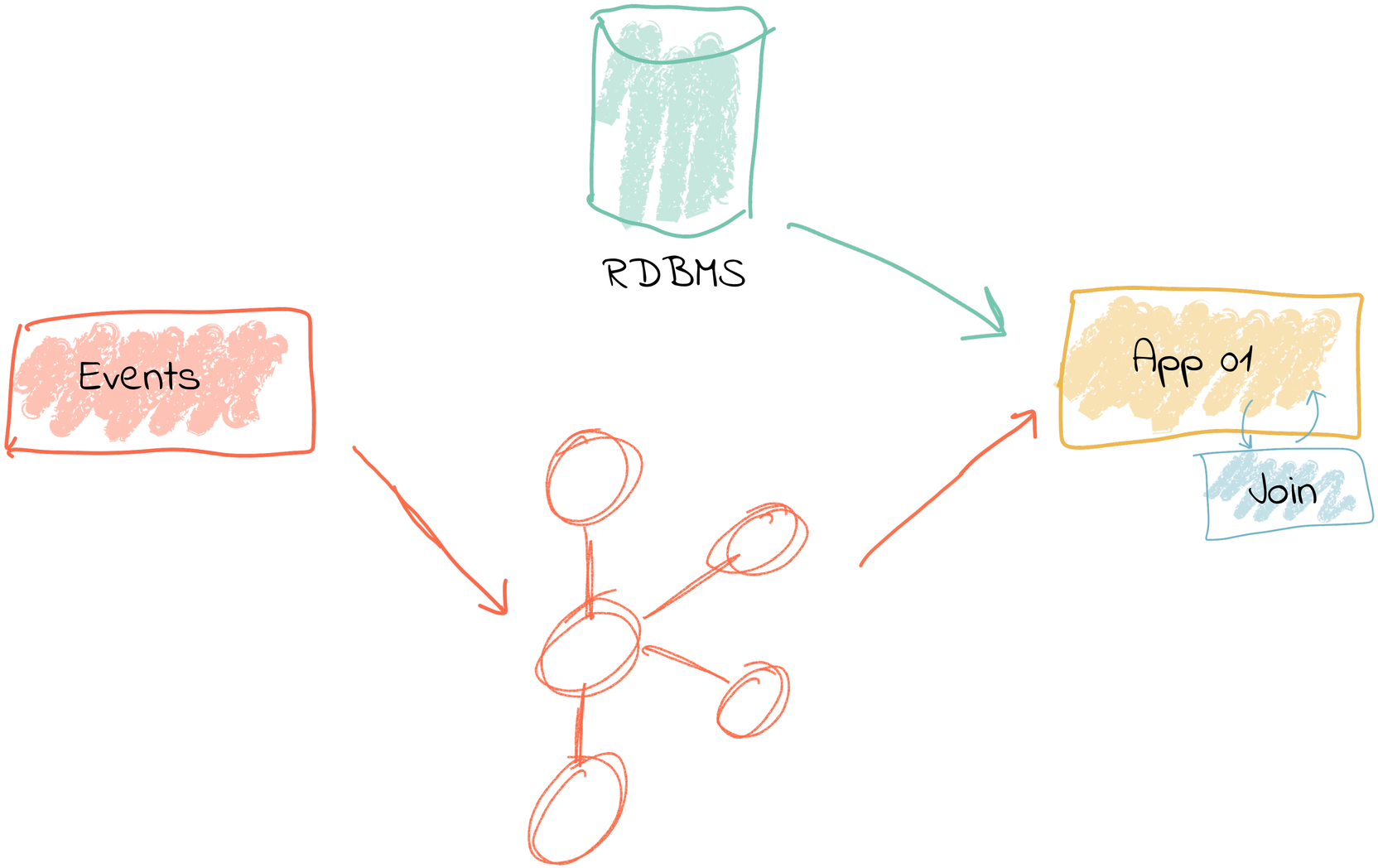
sensor_id	time_epoch	reading
42	1551136074	13.05
42	1551136125	13.11
	1551136125	13.11
42	1551138129	13.04

temp_clean

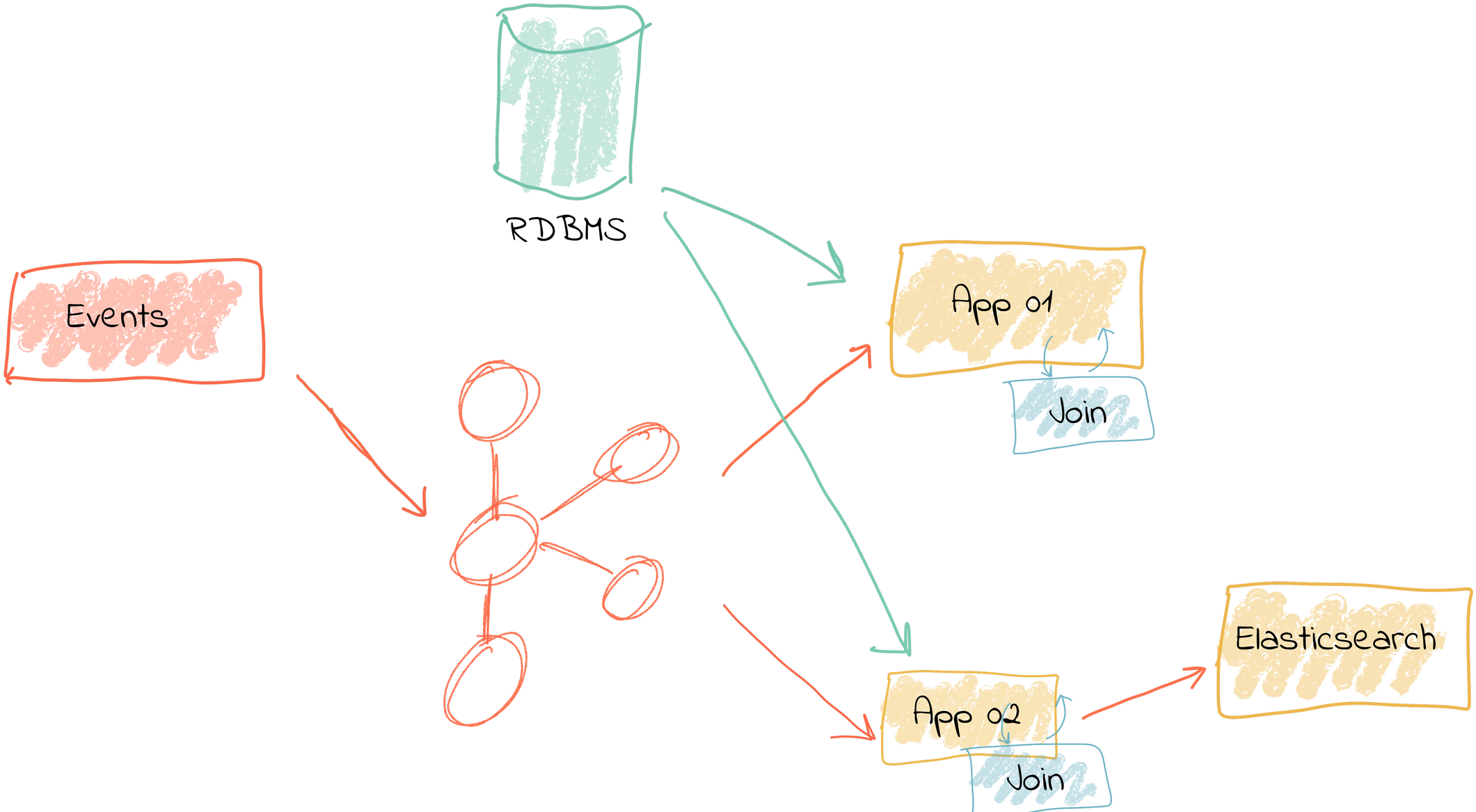
sensor_id	time_epoch	reading
42	1551136074	13.05
42	1551136125	13.11
42	1551138129	13.04



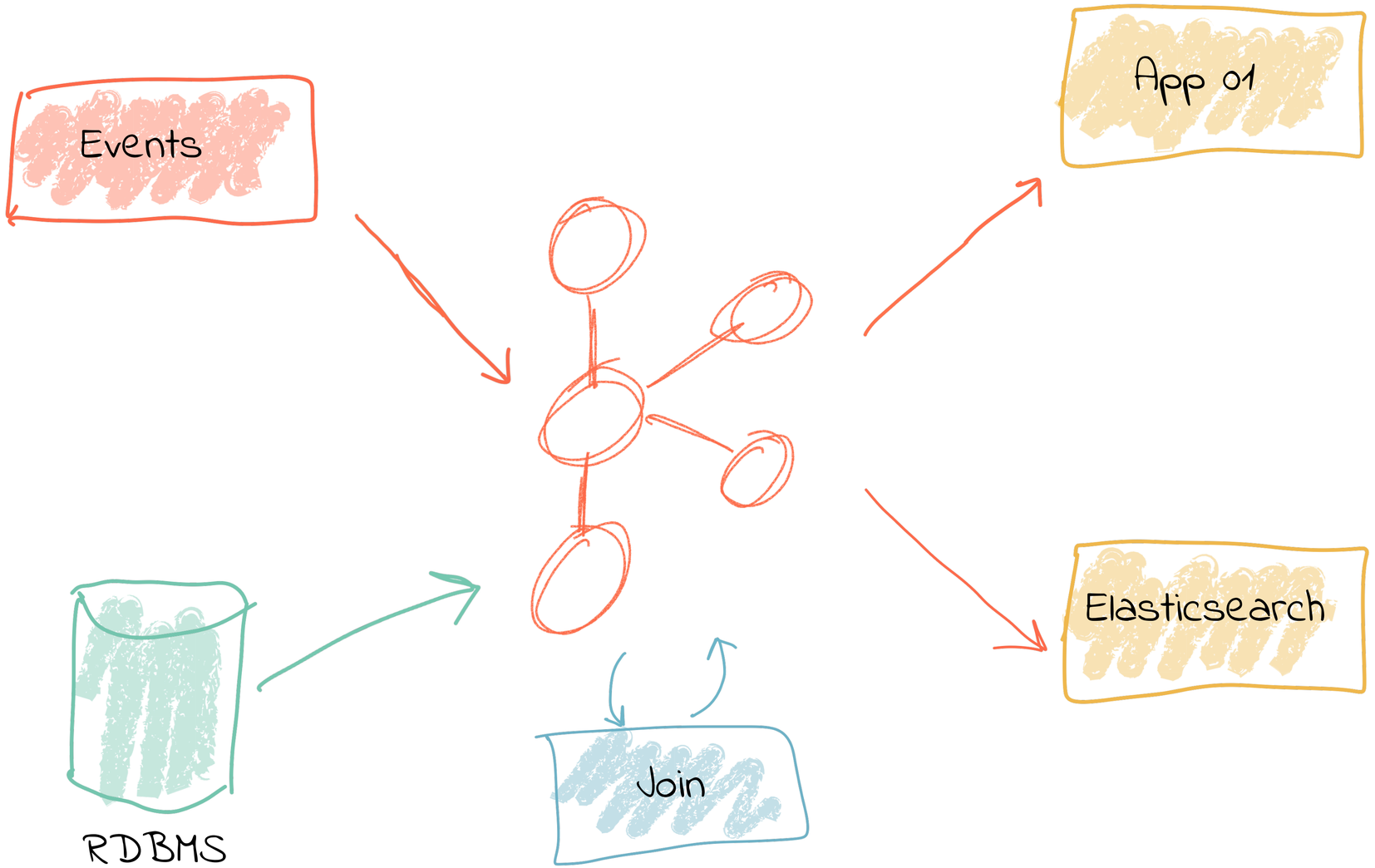
Transform Once, Use Many: Data Enrichment



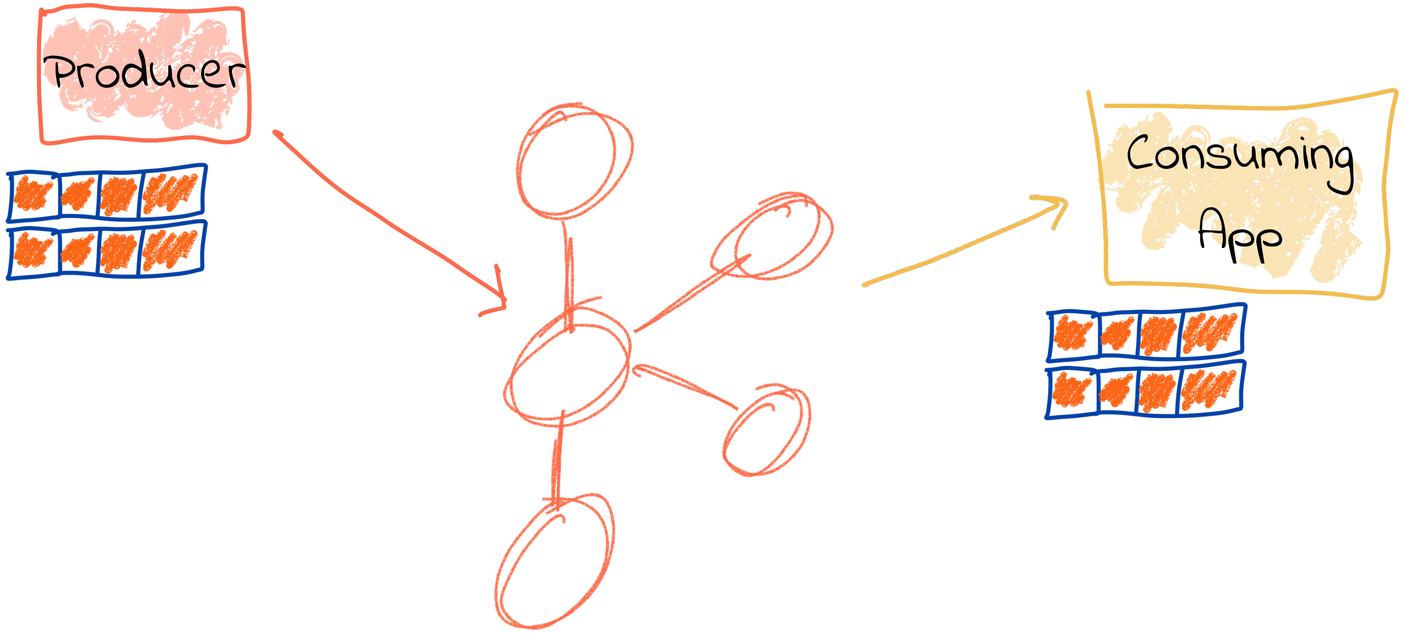
Transform Once, Use Many: Data Enrichment



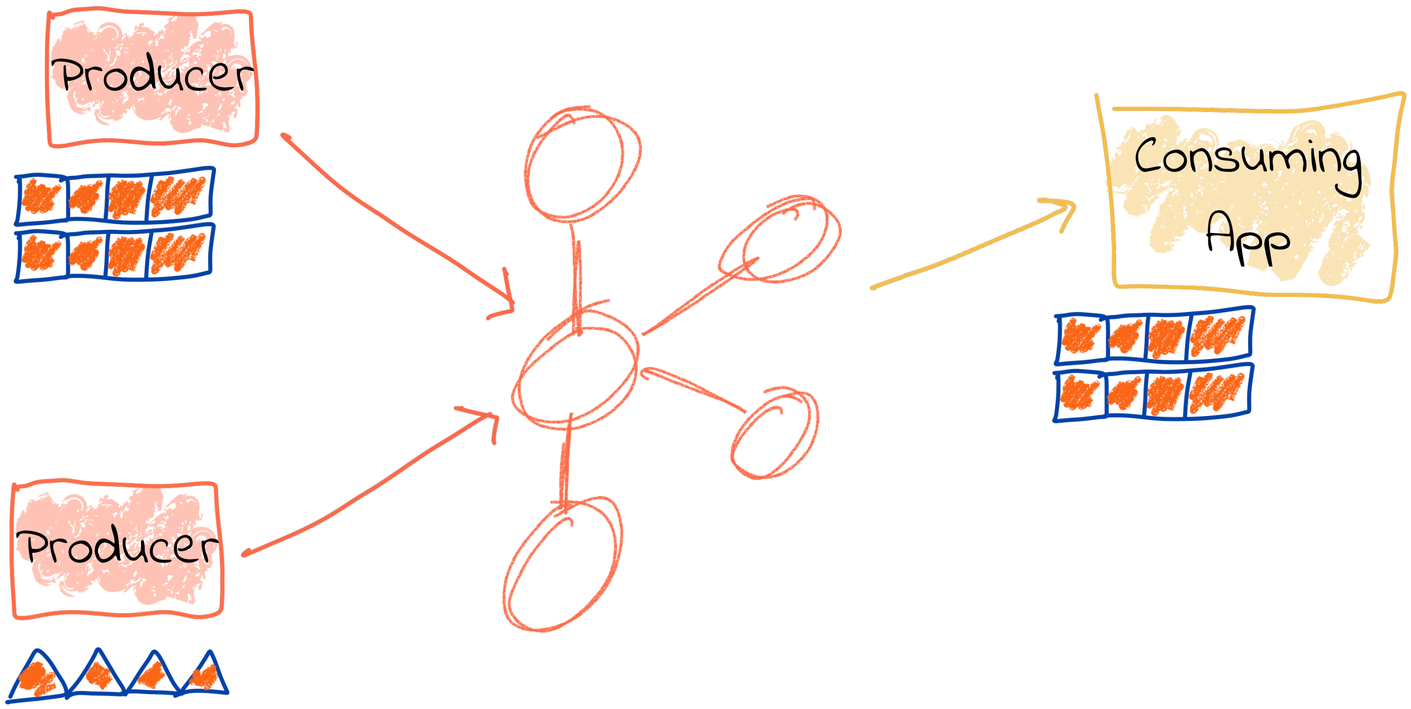
Transform Once, Use Many: Data Enrichment



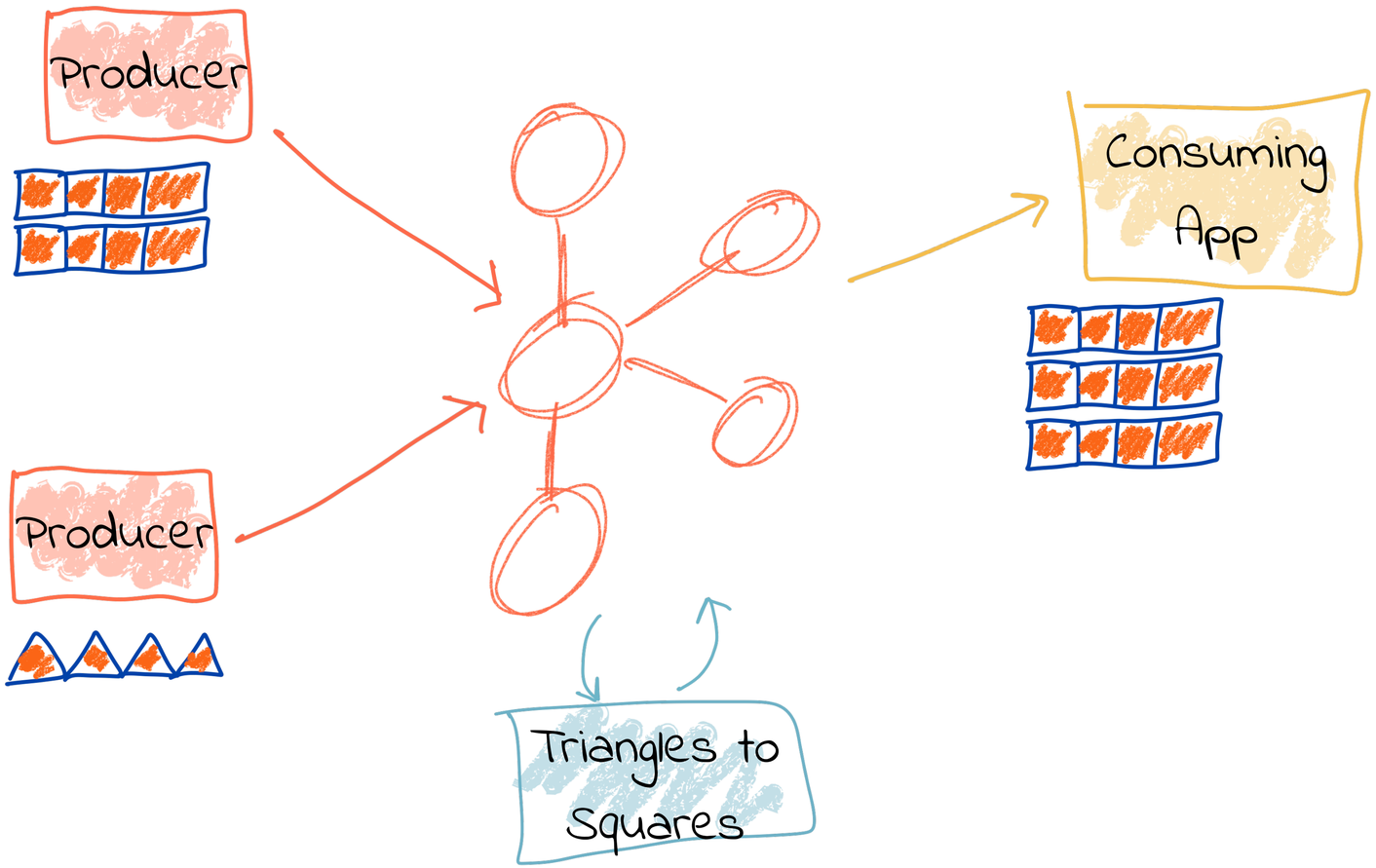
Message Payload Compatibility



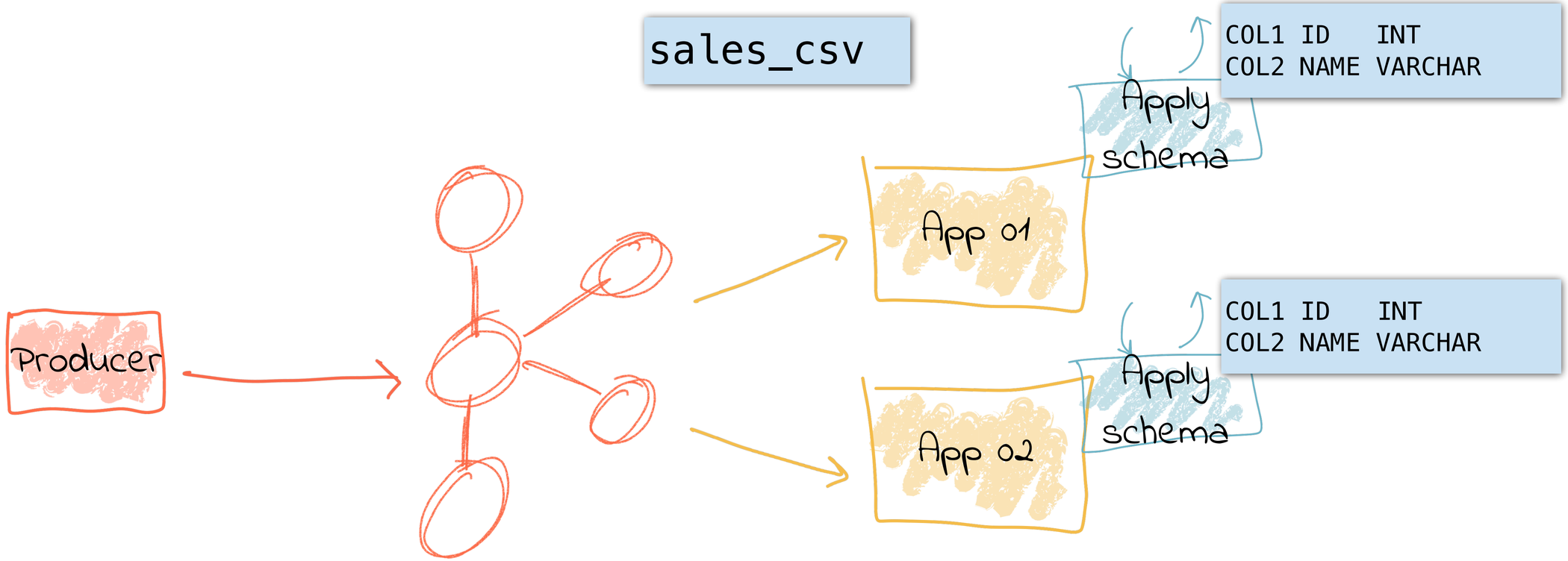
Message Payload Compatibility



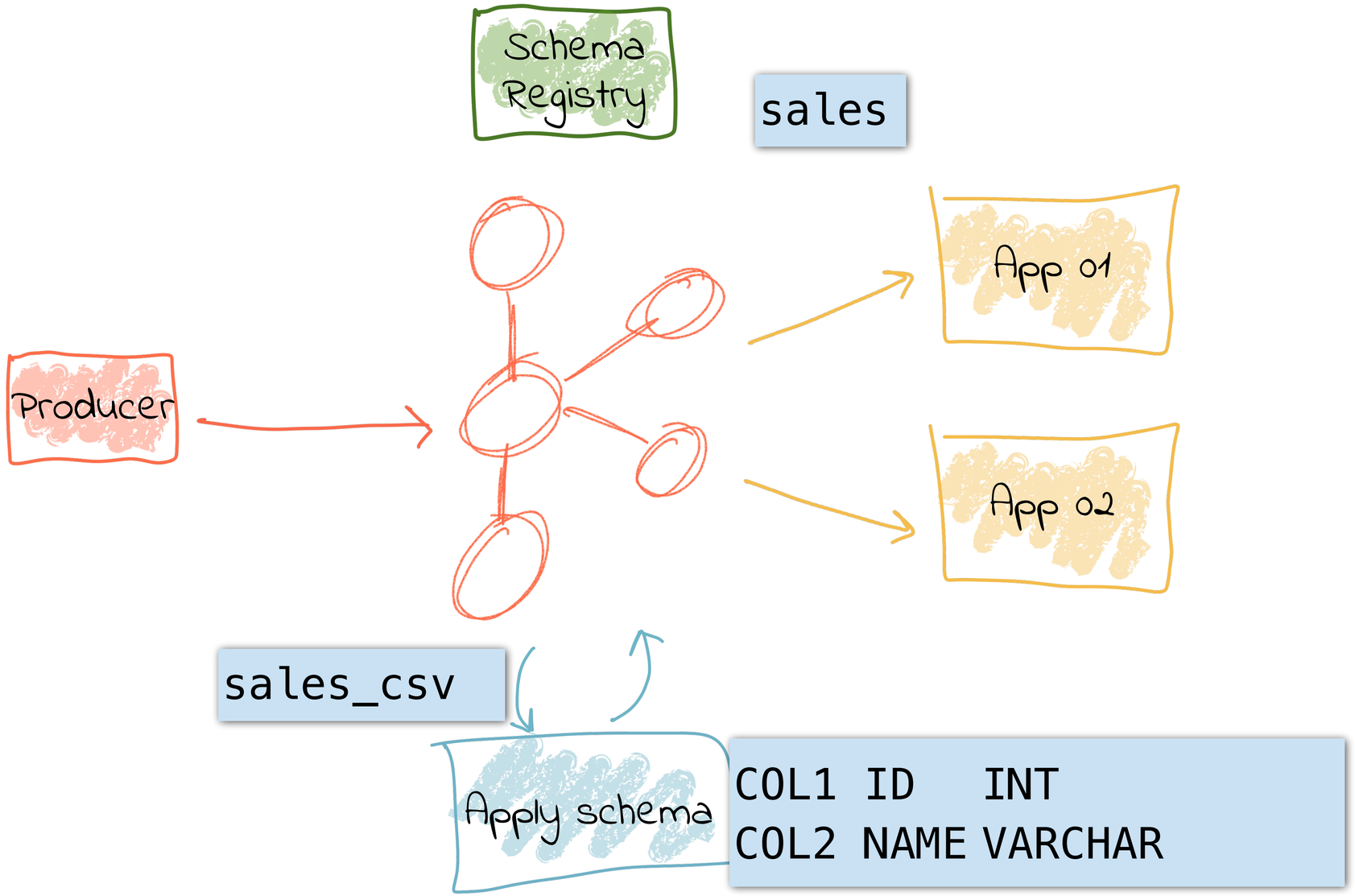
Message Payload Compatibility



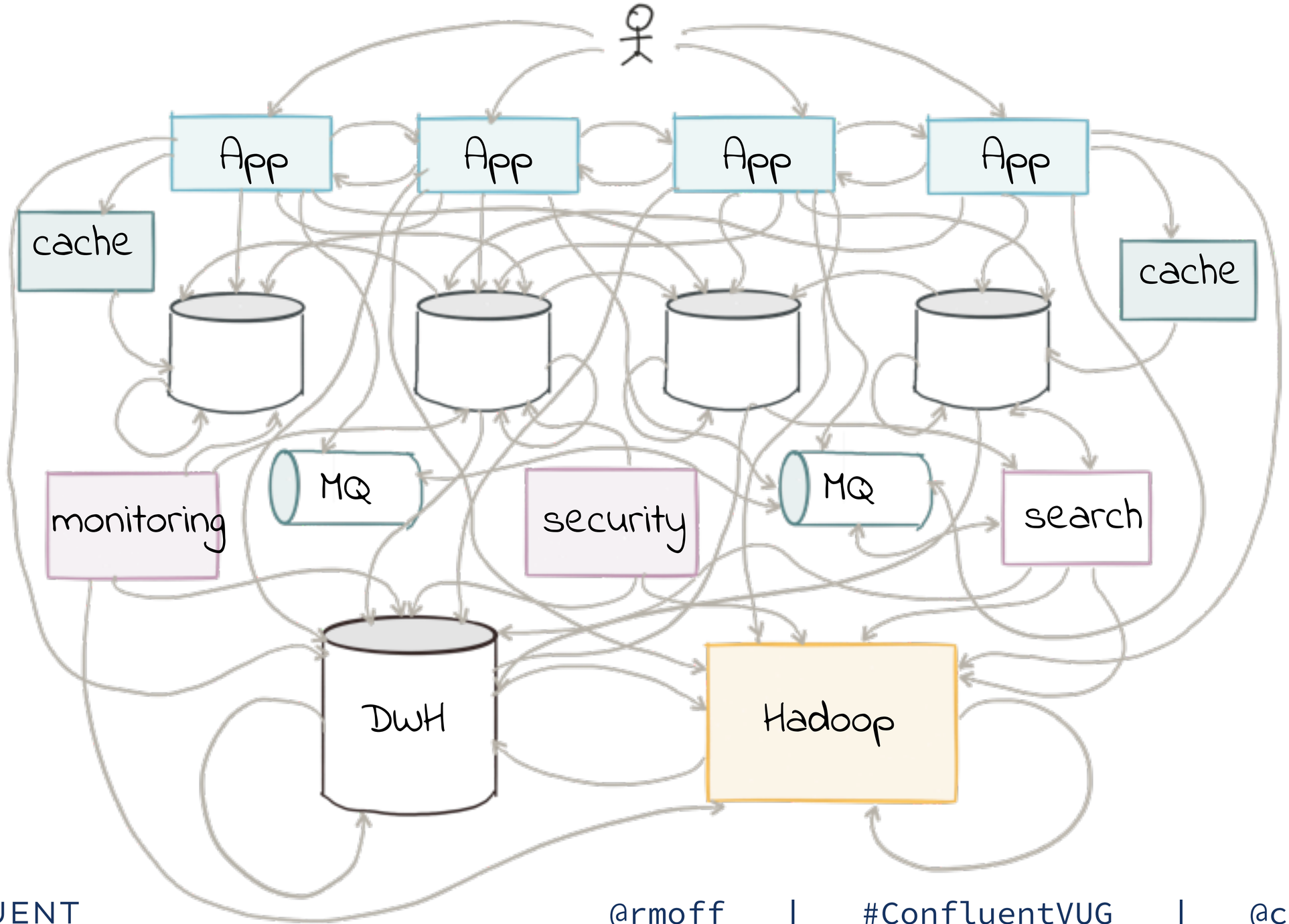
Build Resilient Pipelines with Schemas

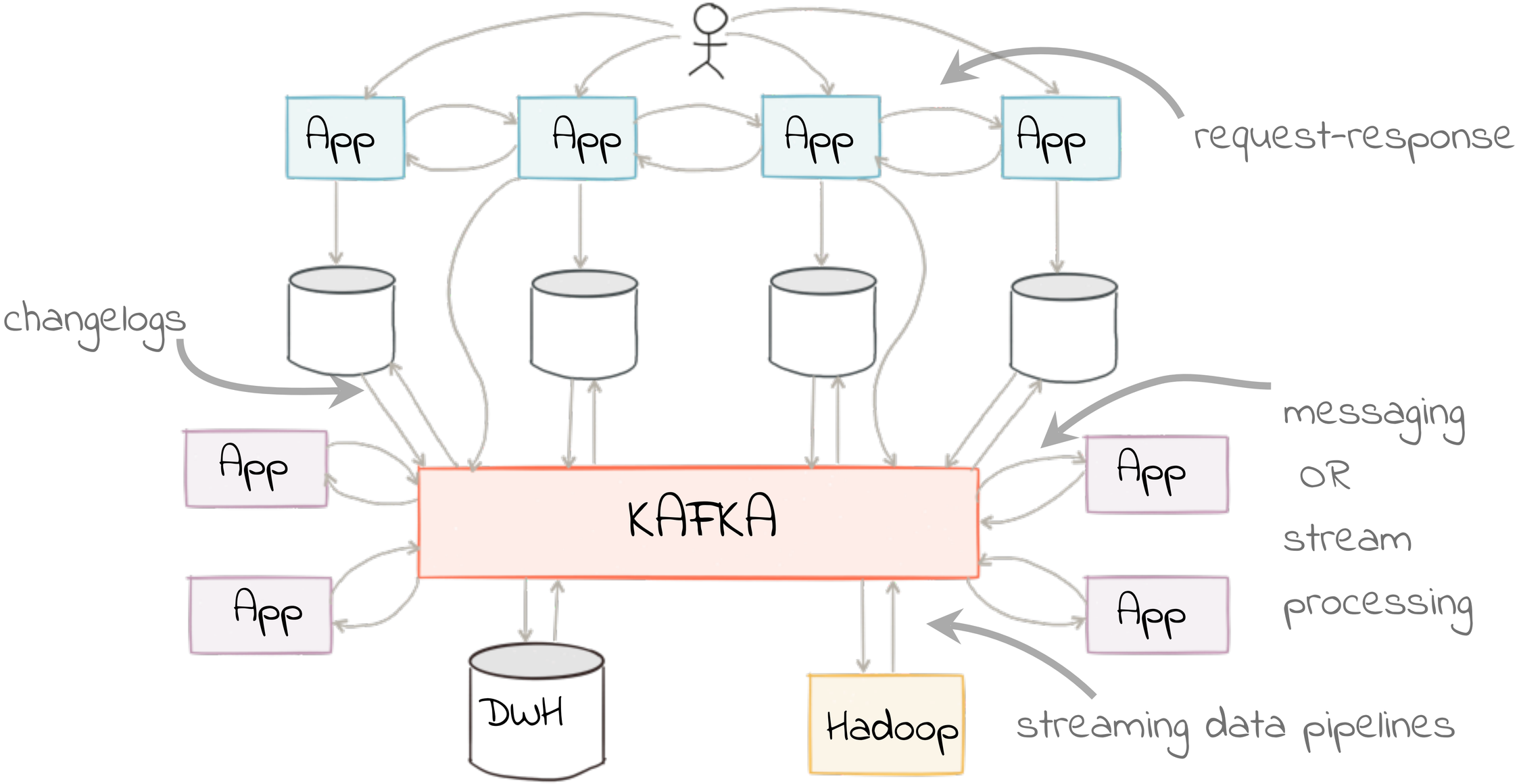


Build Resilient Pipelines with Schemas



Say NO to brittle pipelines
Photo by rmoff


App
App
App
App
cache
cache
MQ
MQ
monitoring
security
search
DWH
Hadoop


App
App
App
App
request-response
changelogs
App
KAFKA
App
messaging
OR
stream
processing
App
App
DWH
Hadoop
streaming data pipelines

Events model the real world

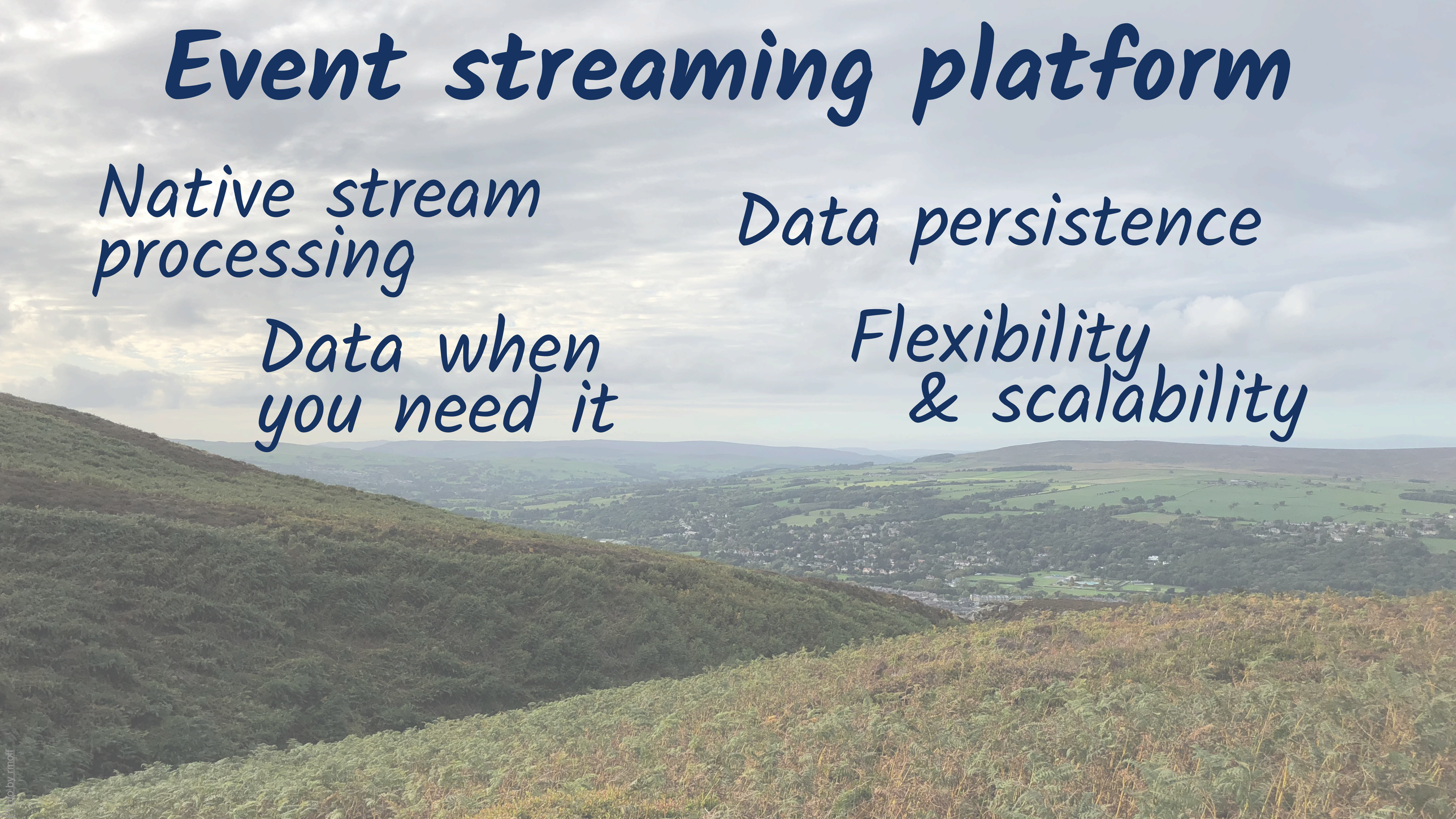
Event streaming platform

Native stream processing

Data persistence

Data when you need it

Flexibility & scalability

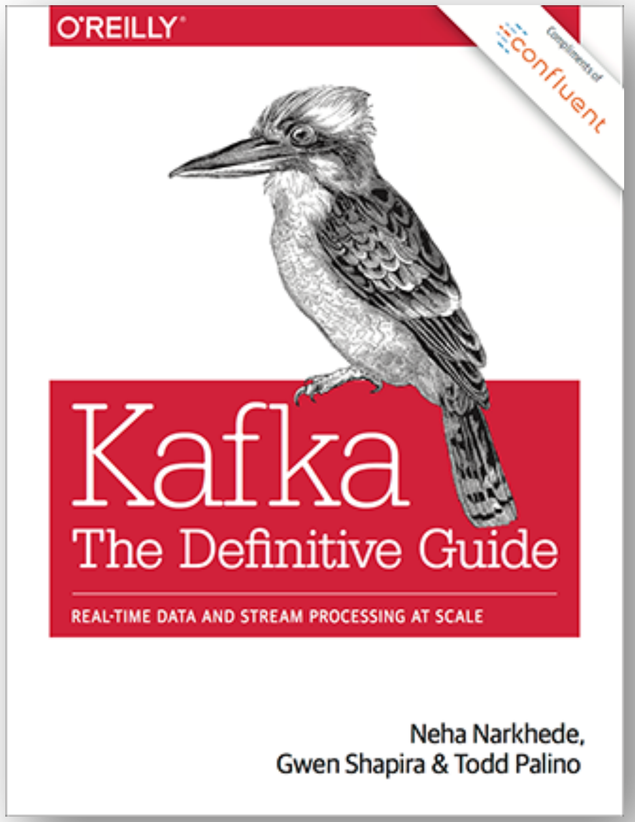


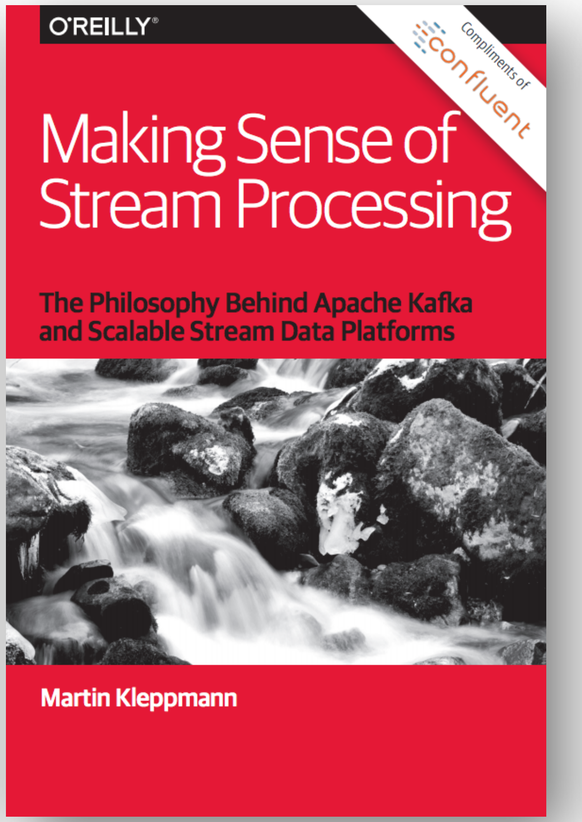
Want to learn more?

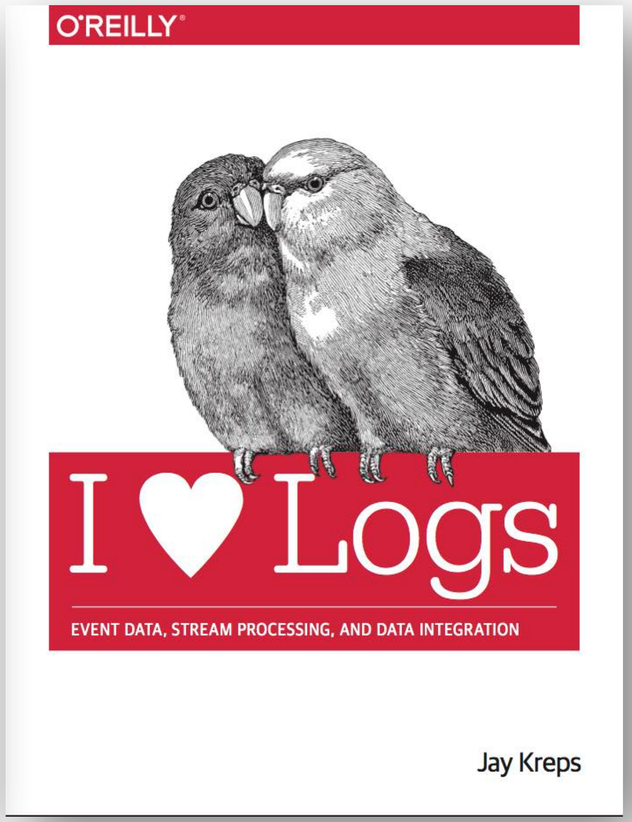
CTAs, not CATs
(sorry, not sorry)

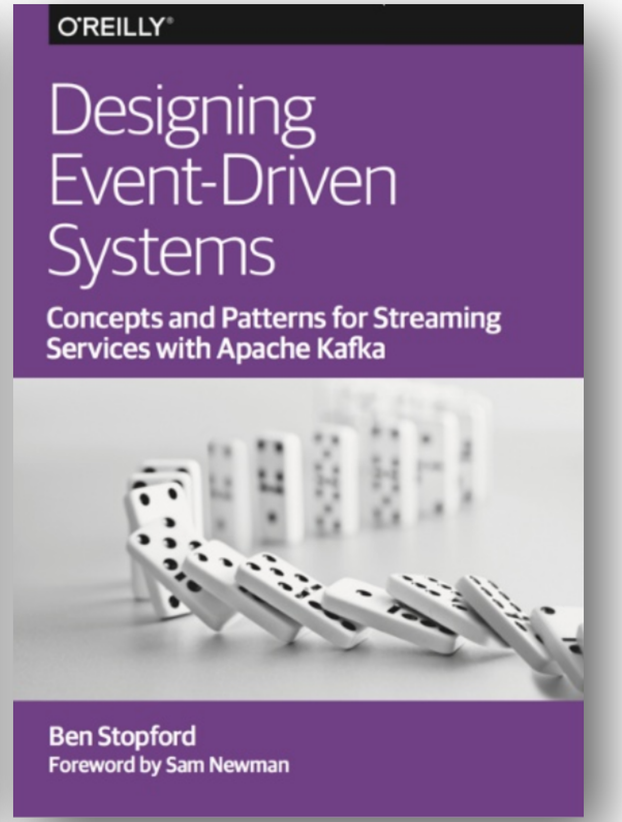
Free Books!

https://rmoff.dev/q2m









60DEVADV

Free money!
(additional $60 towards your bill 😄)

$50 USD off your bill each calendar month for the first three months when you sign up

https://rmoff.dev/ccloud

confluent cloud

Fully Managed Kafka as a Service

* Limited availability. Activate by 11th September 2020. Expires after 90 days of activation. Any unused promo value on the expiration date will be forfeited.

Learn Kafka.
Start building with Apache Kafka at Confluent Developer.
developer.confluent.io

Confluent Community Slack group

cnfl.io/slack

Further reading / watching

- **Kafka as a Platform: the Ecosystem from the Ground Up**
  - https://rmoff.dev/kafka101
- **Apache Kafka and ksqlDB in Action: Let's Build a Streaming Data Pipeline!**
  - https://rmoff.dev/ljc-kafka-01
- **From Zero to Hero with Kafka Connect**
  - https://rmoff.dev/ljc-kafka-02
- **Introduction to ksqlDB**
  - https://rmoff.dev/ljc-kafka-03
- **Integrating Oracle and Kafka**
  - https://rmoff.dev/oracle-and-kafka
- **The Changing Face of ETL: Event-Driven Architectures for Data Engineers**
  - https://rmoff.dev/oredev19-changing-face-of-etl
- **🚂On Track with Apache Kafka: Building a Streaming Platform solution with Rail Data**
  - https://rmoff.dev/oredev19-on-track-with-kafka

#EOF